FIG.1

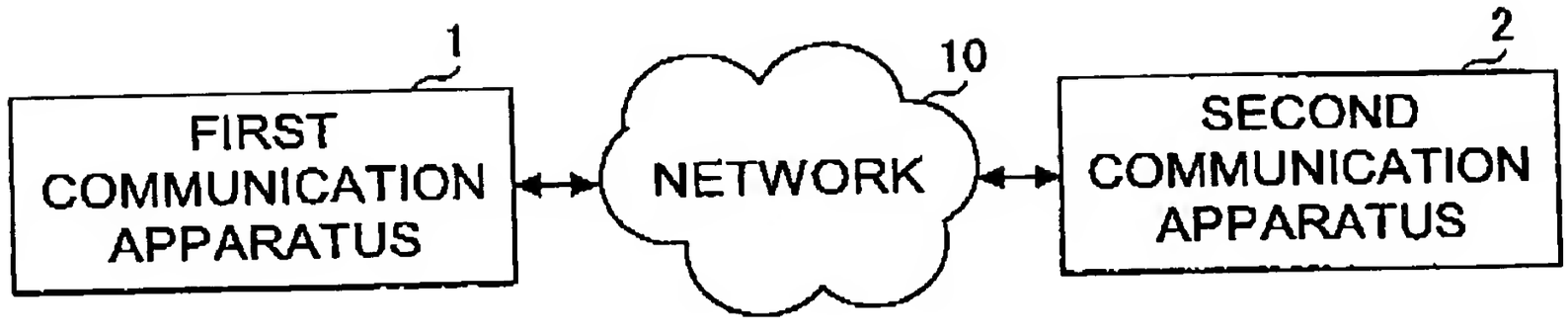


FIG.2A

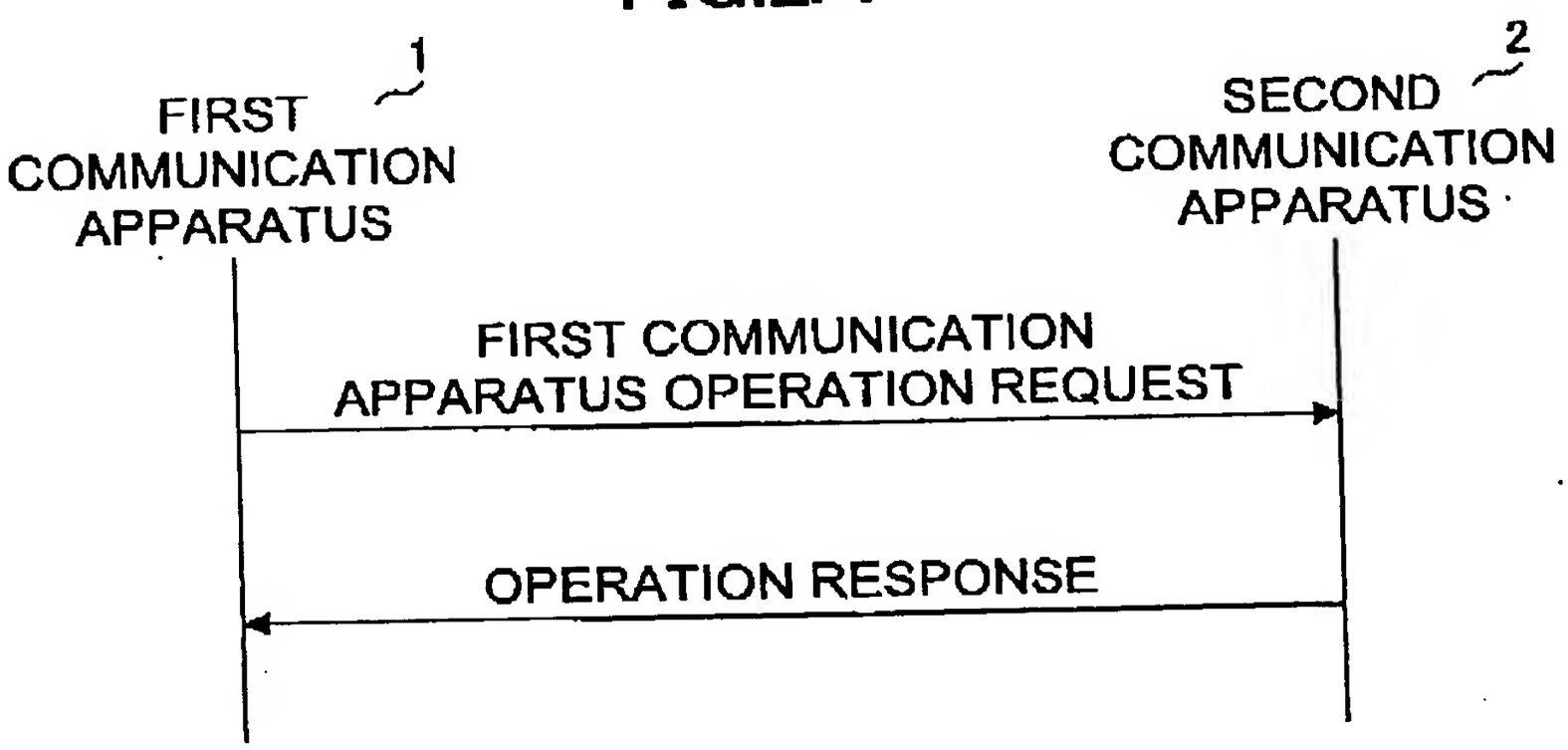


FIG.2B

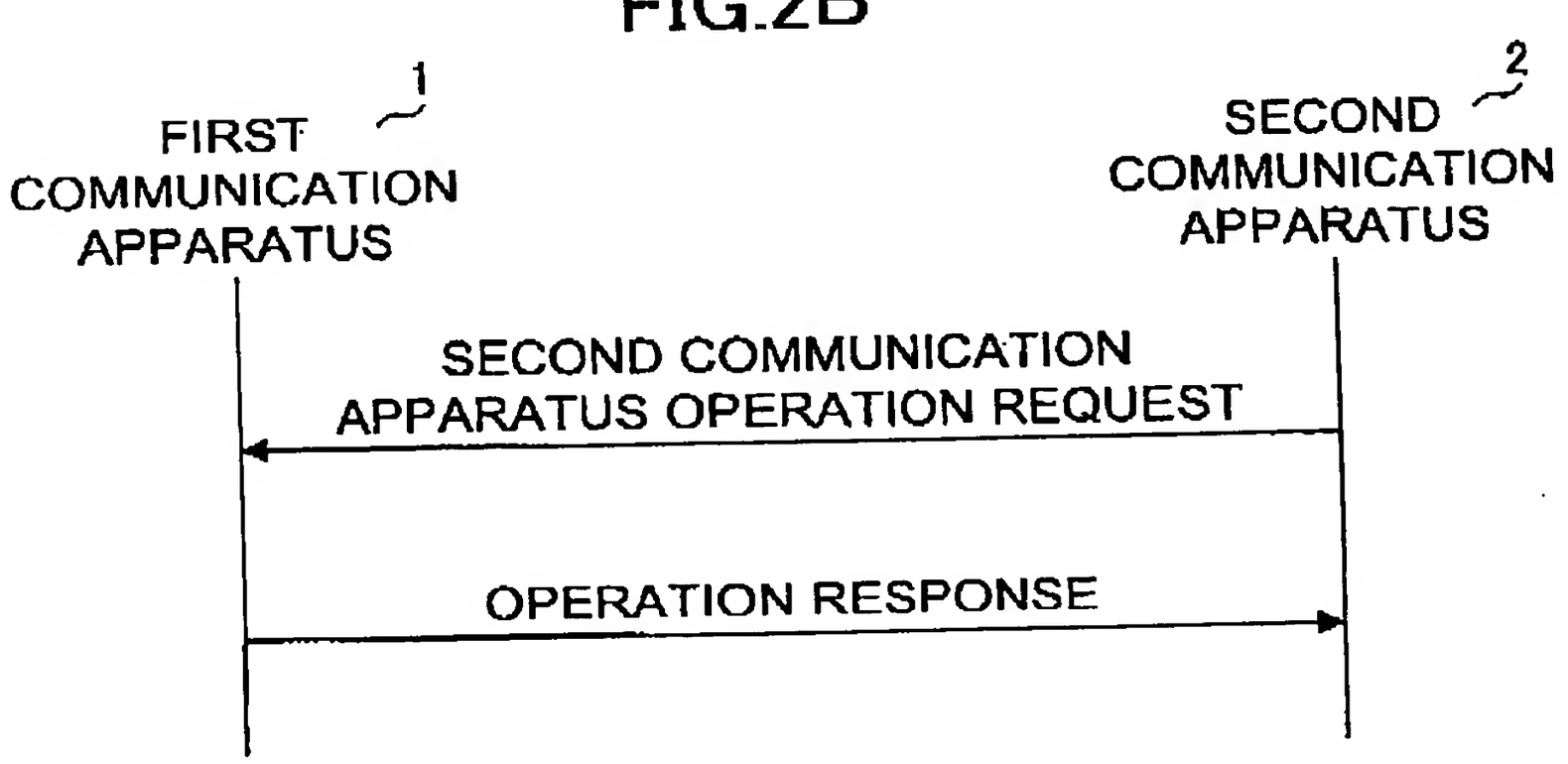


FIG.3

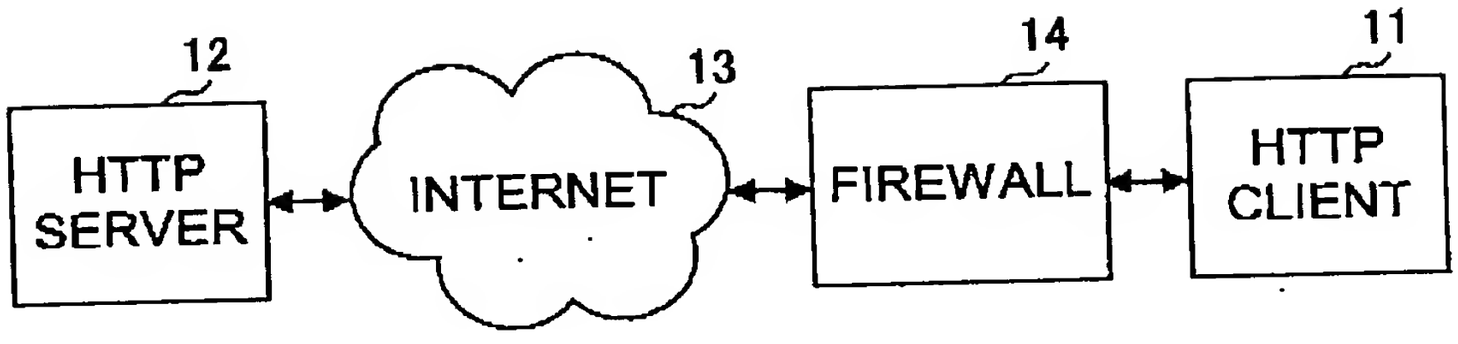


FIG.4A

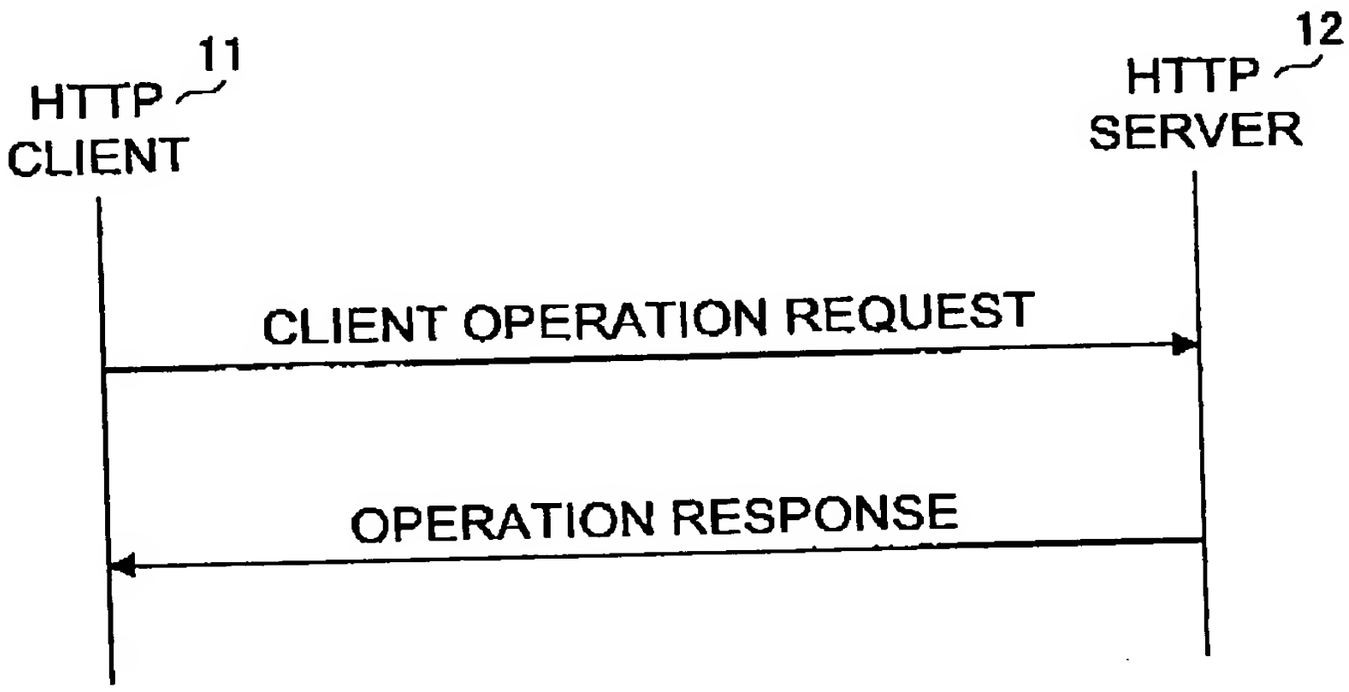


FIG.4B

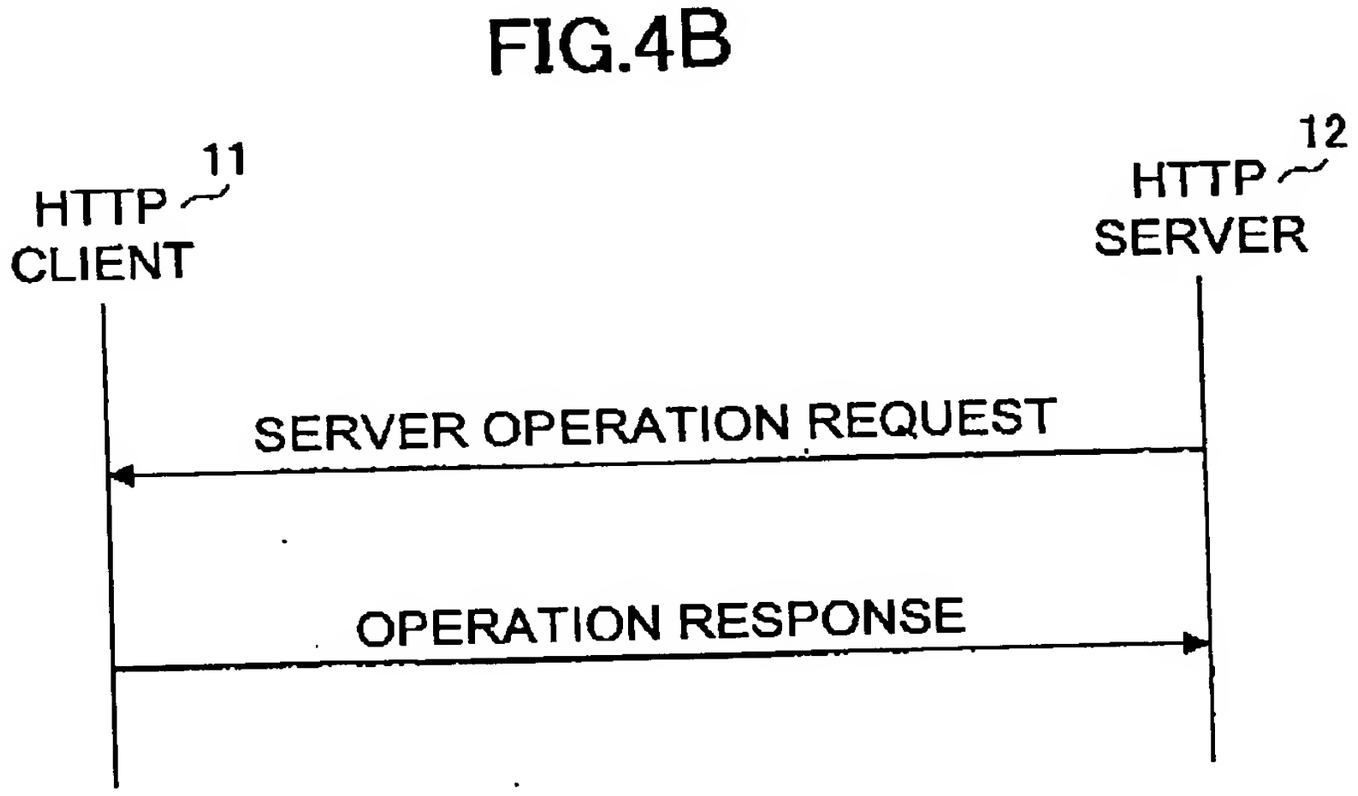


FIG.5

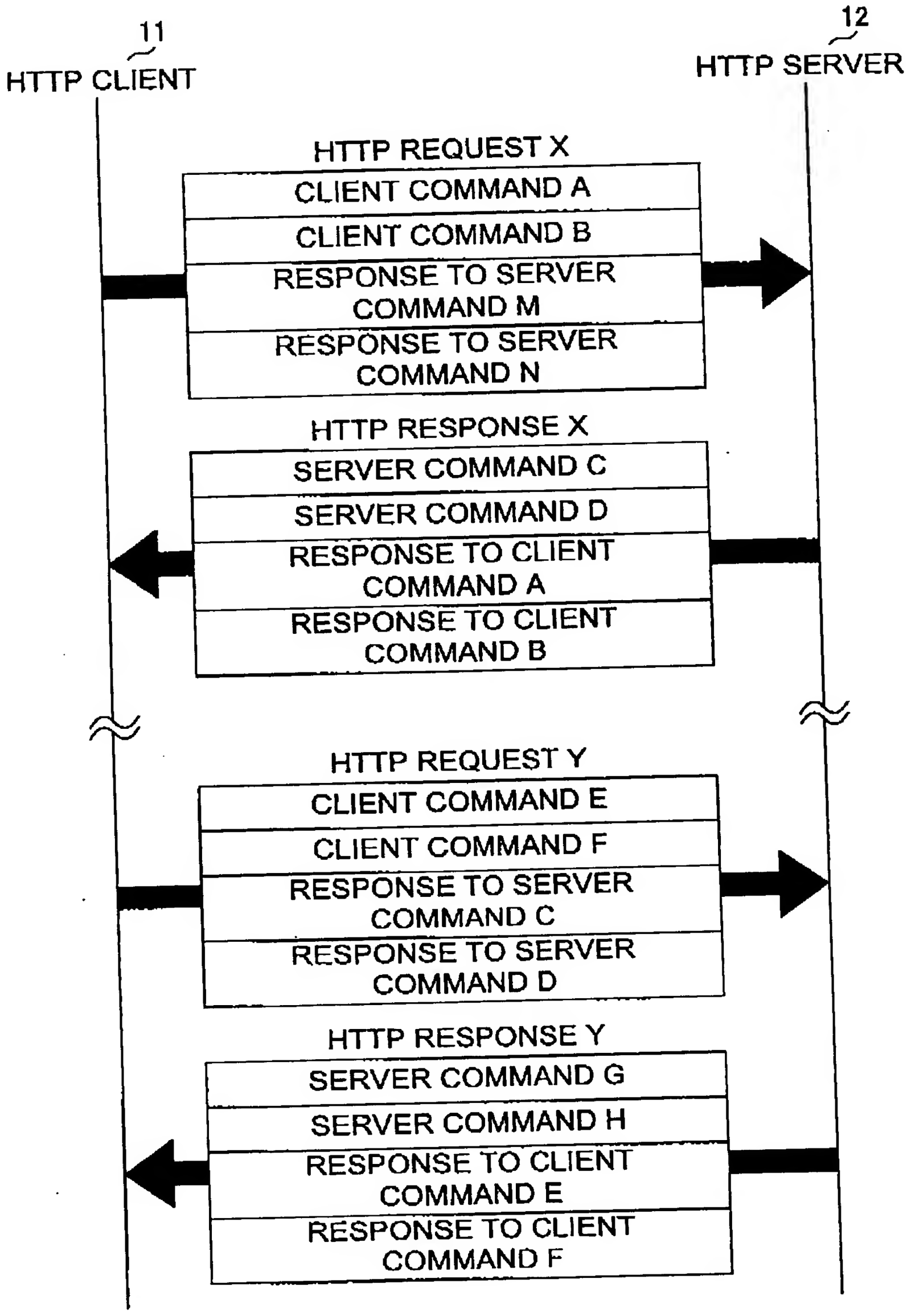

11
HTTP CLIENT
12
HTTP SERVER
HTTP REQUEST X
CLIENT COMMAND A
CLIENT COMMAND B
RESPONSE TO SERVER COMMAND M
RESPONSE TO SERVER COMMAND N
HTTP RESPONSE X
SERVER COMMAND C
SERVER COMMAND D
RESPONSE TO CLIENT COMMAND A
RESPONSE TO CLIENT COMMAND B
HTTP REQUEST Y
CLIENT COMMAND E
CLIENT COMMAND F
RESPONSE TO SERVER COMMAND C
RESPONSE TO SERVER COMMAND D
HTTP RESPONSE Y
SERVER COMMAND G
SERVER COMMAND H
RESPONSE TO CLIENT COMMAND E
RESPONSE TO CLIENT COMMAND F

FIG.6

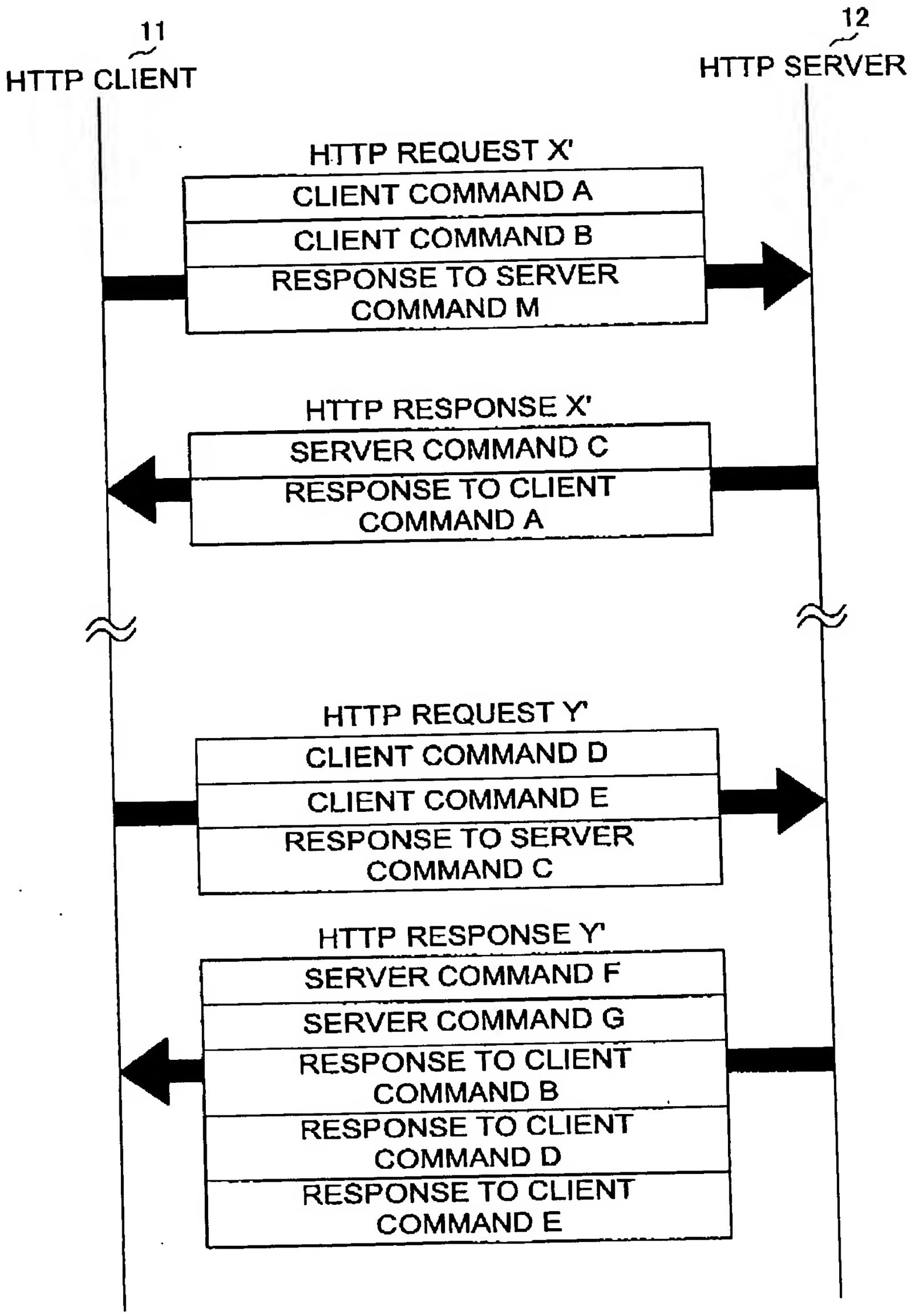

11
HTTP CLIENT
12
HTTP SERVER
HTTP REQUEST X'
CLIENT COMMAND A
CLIENT COMMAND B
RESPONSE TO SERVER COMMAND M
HTTP RESPONSE X'
SERVER COMMAND C
RESPONSE TO CLIENT COMMAND A
HTTP REQUEST Y'
CLIENT COMMAND D
CLIENT COMMAND E
RESPONSE TO SERVER COMMAND C
HTTP RESPONSE Y'
SERVER COMMAND F
SERVER COMMAND G
RESPONSE TO CLIENT COMMAND B
RESPONSE TO CLIENT COMMAND D
RESPONSE TO CLIENT COMMAND E

FIG.7

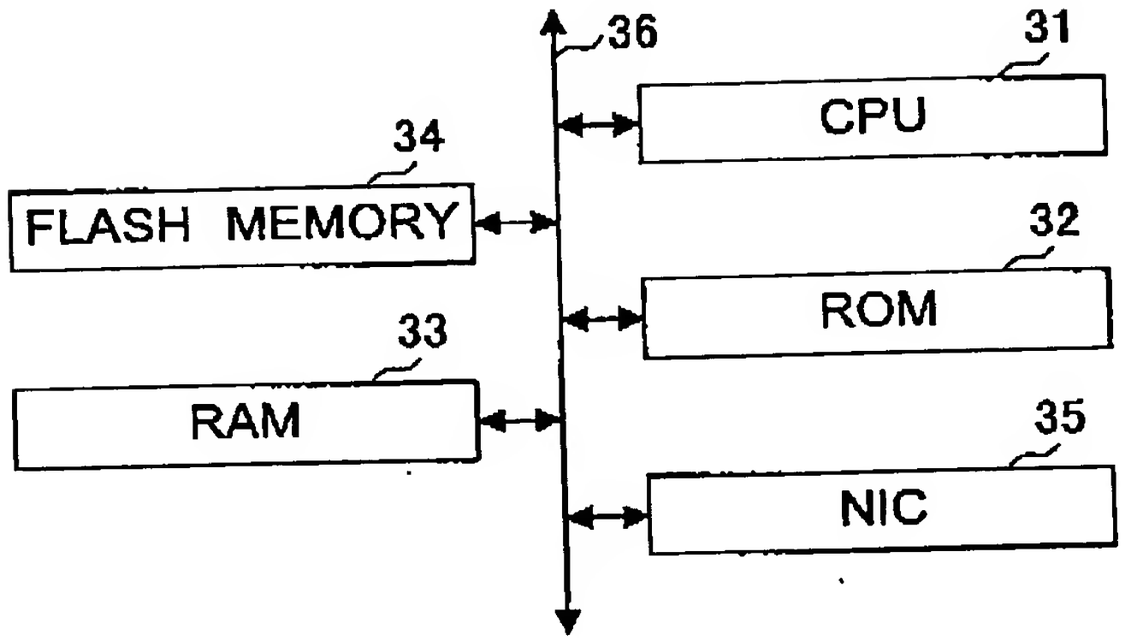
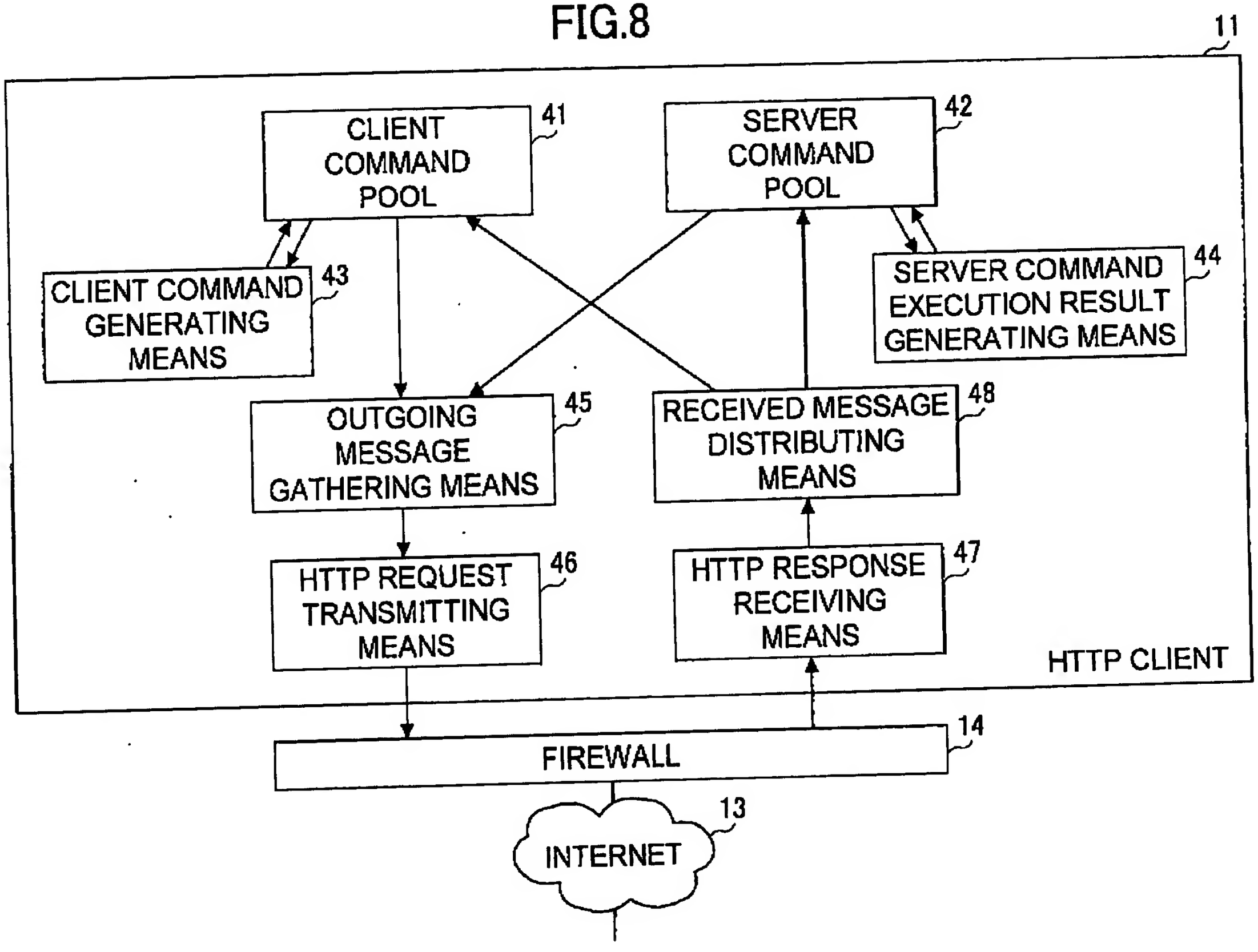


FIG.8



OBLON, SPIVAK, ET AL
DOCKET #: 243019US2
INV: Hiroyuki MATSUSHIMA
SHEET 6 OF 31

FIG.9

CLIENT COMMAND SHEET IN
HTTP CLIENT

COMMAND ID
METHOD NAME (e.g., TROUBLE NOTIFICATION)
INPUT PARAMETER (e.g., TROUBLE CONTENT)
STATUS (INITIAL VALUE: NOT SENT)
CLIENT COMMAND EXECUTION RESULT NOTIFYING DESTINATION
OUTPUT PARAMETER (BLANK UNTIL RESPONSE IS ACQUIRED)

FIG.10

SERVER COMMAND SHEET IN
HTTP CLIENT

COMMAND ID
METHOD NAME (e.g., COUNTER ACQUISITION)
INPUT PARAMETER
STATUS (INITIAL VALUE: NOT SENT)
OUTPUT PARAMETER (BLANK UNTIL PROCESS IS COMPLETED)
SERVER COMMAND NOTIFYING DESTINATION

FIG.11

HTTP REQUEST

```
POST /aaa HTTP/1.1
Content-Type:multipart/mixed;boundary=MIME_boundary
Content-Length:nnnn

--MIME_boundary
Content-Type:text/xml;charset=UTF-8
Content-Transfer-Encoding:8bit
SOAPAction:"SOAP Action URI"
X-SOAP-Type:Requset

<s:Envelope>
   <--SOAP Request-->
</s:Envelope>

--MIME_boundary
Content-Type:text/xml;charset=UTF-8
Content-Transfer-Encoding:8bit
X-SOAP-Type:Response

<s:Envelope>
   <--SOAP Response-->
</s:Envelope>

--MIME_boundary
Content-Type:text/xml;charset=UTF-8
Content-Transfer-Encoding:8bit
X-SOAP-Type:Response

<s:Envelope>
   <--SOAP Response-->
</s:Envelope>

--MIME_boundary
Content-Type:text/xml;charset=UTF-8
Content-Transfer-Encoding:8bit
X-SOAP-Type:Response

<s:Envelope>
   <--SOAP Response-->
</s:Envelope>

--MINE_boundary--
```

PART 1

PART 2

PART 3

PART 4

FIG.12

HTTP RESPONSE

```
HTTP/1.1  200  OK
Content-Type:multipart/mixed;boundary=MIME_boundary
Content-Length:nnnn

--MIME_boundary
Content-Type:text/xml;charset=UTF-8
Content-Transfer-Encoding:8bit
SOAPAction:"SOAP Action URI"
X-SOAP-Type:Requset

<s:Envelope>
   <--SOAP Request-->
</s:Envelope>

--MIME_boundary
Content-Type:text/xml;charset=UTF-8
Content-Transfer-Encoding:8bit
X-SOAP-Type:Response

<s:Envelope>
   <--SOAP Response-->
</s:Envelope>

--MIME_boundary
Content-Type:text/xml;charset=UTF-8
Content-Transfer-Encoding:8bit
X-SOAP-Type:Response

<s:Envelope>
   <--SOAP Response-->
</s:Envelope>

--MIME_boundary
Content-Type:text/xml;charset=UTF-8
Content-Transfer-Encoding:8bit
X-SOAP-Type:Response

<s:Envelope>
   <--SOAP Response-->
</s:Envelope>

--MINE_boundary--
```

PART 1

PART 2

PART 3

PART 4

FIG.13

EXEMPLARY PART DESCRIBING CLIENT COMMAND

```
Content-Type: text/xml; charset=UTF-8
Content-Transfer-Encoding: 8bit
X-SOAP-Type : Request
SOAPAction: "http://www.foo.com/ server/ errorNotification"

<s:Envelope
 xmlns:s="http://schemas.xmlsoap.org/soap/envelope/"
 xmlns:se="http://schemas.xmlsoap.org/soap/encoding/"
 xmlns:n="http://www.foo.com/header"
 xmlns:ns="http://www.foo.com/server"
 s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding">


 <s:Header>
    <n:Request ID> 12345 </n:Request ID>
 </s:Header>
  <s:Body>
   <ns:Trouble Notification>
       <ErrorID> 1111 </ErrorID>
       <Description>Hard Disk Drive Trouble </Description>
   </ns:Trouble Notification>
  </s:Body>
</s:Envelope>
```

FIG.14

EXEMPLARY PART DESCRIBING RESPONSE TO CLIENT COMMAND

```
Content-Type: text/xml; charset=UTF-8
Content-Transfer-Encoding: 8bit
X-SOAP-Type : Response


<s:Envelope
 xmlns:s="http://schemas.xmlsoap.org/soap/envelope/"
 xmlns:se="http://schemas.xmlsoap.org/soap/encoding/"
 xmlns:n="http://www.foo.com/header"
 xmlns:ns="http://www.foo.com/server"
 s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding">


 <s:Header>
   <n:Request ID> 12345 </n:Request ID>
 </s:Header>
 <s:Body>
  <ns:Trouble Notification Response>
     <Reception Result> OK </Reception Result>
  </ns:Trouble Notification Response>
 </s:Body>
</s:Envelope>
```

FIG.15

EXEMPLARY PART DESCRIBING SERVER COMMAND

```
Content-Type: text/xml; charset=UTF-8
Content-Transfer-Encoding: 8bit
X-SOAP-Type : Request
SOAPAction: "http://www.foo.com/ client/getTemperature"

<s:Envelope
 xmlns:s="http://schemas.xmlsoap.org/soap/envelope/"
 xmlns:se="http://schemas.xmlsoap.org/soap/encoding/"
 xmlns:n="http://www.foo.com/header"
 xmlns:nc="http://www.foo.com/client"
 s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding">


 <s:Header>
    <n:Request ID> 98765 </n:Request ID>
 </s:Header>
  <s:Body>
   <nc:Temperature Sensor Value Acqisition>
       <SensorID> 3 </SensorID>
   </nc:Temperature Sensor>
  </s:Body>
</s:Envelope>
```

FIG.16

EXEMPLARY PART DESCRIBING RESPONSE TO SERVER COMMAND

```
Content-Type: text/xml; charset=UTF-8
Content-Transfer-Encoding: 8bit
X-SOAP-Type : Response


<s:Envelope
 xmlns:s="http://schemas.xmlsoap.org/soap/envelope/"
 xmlns:se="http://schemas.xmlsoap.org/soap/encoding/"
 xmlns:n="http://www.foo.com/header"
 xmlns:nc="http://www.foo.com/client"
 s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding">


 <s:Header>
    <n:Request ID> 98765 </n:Request ID>
 </s:Header>
  <s:Body>
   <nc:Temperature Sensor Value Acqisition Response>
       <Temperature> 52 </Temperature>
   </nc:Temperature Sensor Value Acqisition Response>
  </s:Body>
</s:Envelope>
```

FIG.17

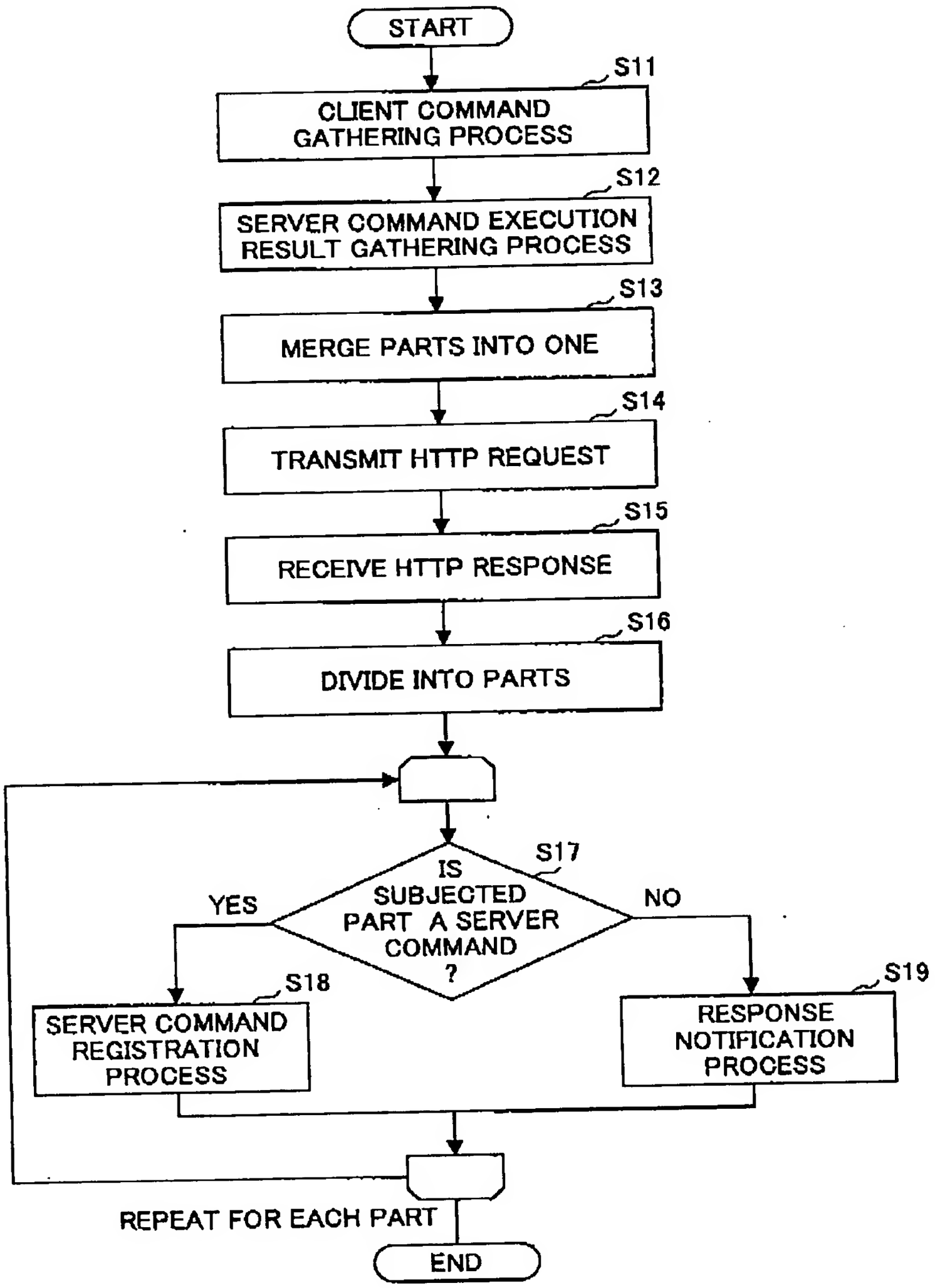

START
S11
CLIENT COMMAND GATHERING PROCESS
S12
SERVER COMMAND EXECUTION RESULT GATHERING PROCESS
S13
MERGE PARTS INTO ONE
S14
TRANSMIT HTTP REQUEST
S15
RECEIVE HTTP RESPONSE
S16
DIVIDE INTO PARTS
S17
IS SUBJECTED PART A SERVER COMMAND ?
YES
NO
S18
SERVER COMMAND REGISTRATION PROCESS
S19
RESPONSE NOTIFICATION PROCESS
REPEAT FOR EACH PART
END

FIG.18

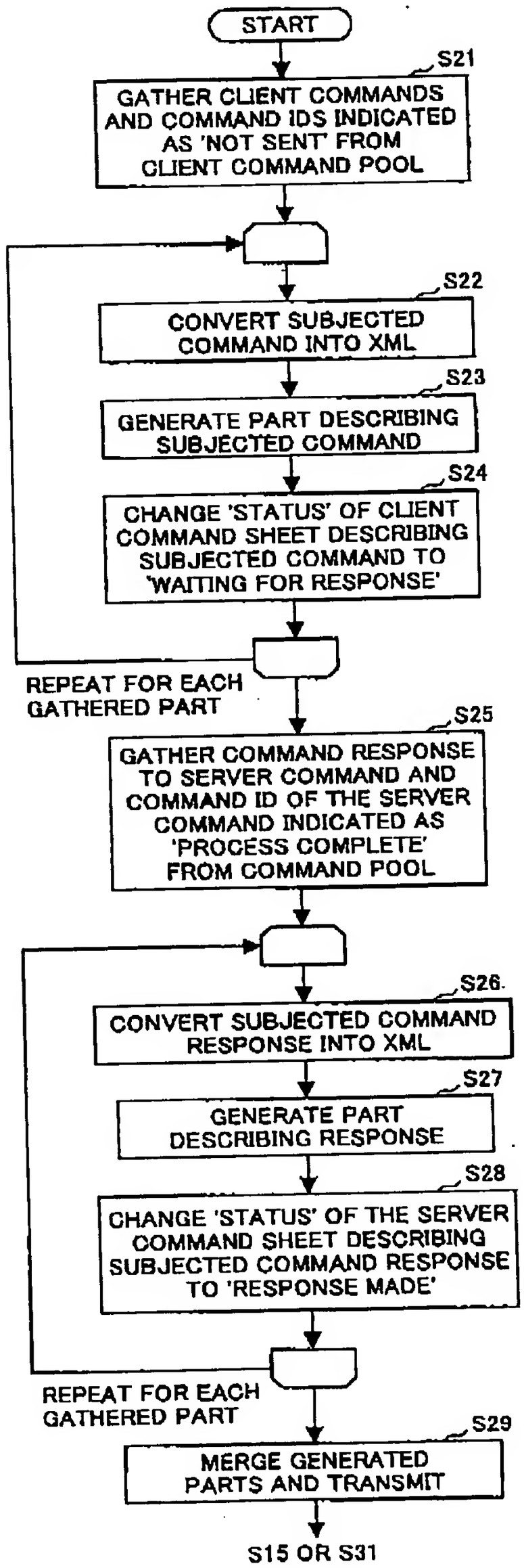

START
S21
GATHER CLIENT COMMANDS AND COMMAND IDS INDICATED AS 'NOT SENT' FROM CLIENT COMMAND POOL
S22
CONVERT SUBJECTED COMMAND INTO XML
S23
GENERATE PART DESCRIBING SUBJECTED COMMAND
S24
CHANGE 'STATUS' OF CLIENT COMMAND SHEET DESCRIBING SUBJECTED COMMAND TO 'WAITING FOR RESPONSE'
REPEAT FOR EACH GATHERED PART
S25
GATHER COMMAND RESPONSE TO SERVER COMMAND AND COMMAND ID OF THE SERVER COMMAND INDICATED AS 'PROCESS COMPLETE' FROM COMMAND POOL
S26
CONVERT SUBJECTED COMMAND RESPONSE INTO XML
S27
GENERATE PART DESCRIBING RESPONSE
S28
CHANGE 'STATUS' OF THE SERVER COMMAND SHEET DESCRIBING SUBJECTED COMMAND RESPONSE TO 'RESPONSE MADE'
REPEAT FOR EACH GATHERED PART
S29
MERGE GENERATED PARTS AND TRANSMIT
S15 OR S31

FIG.19

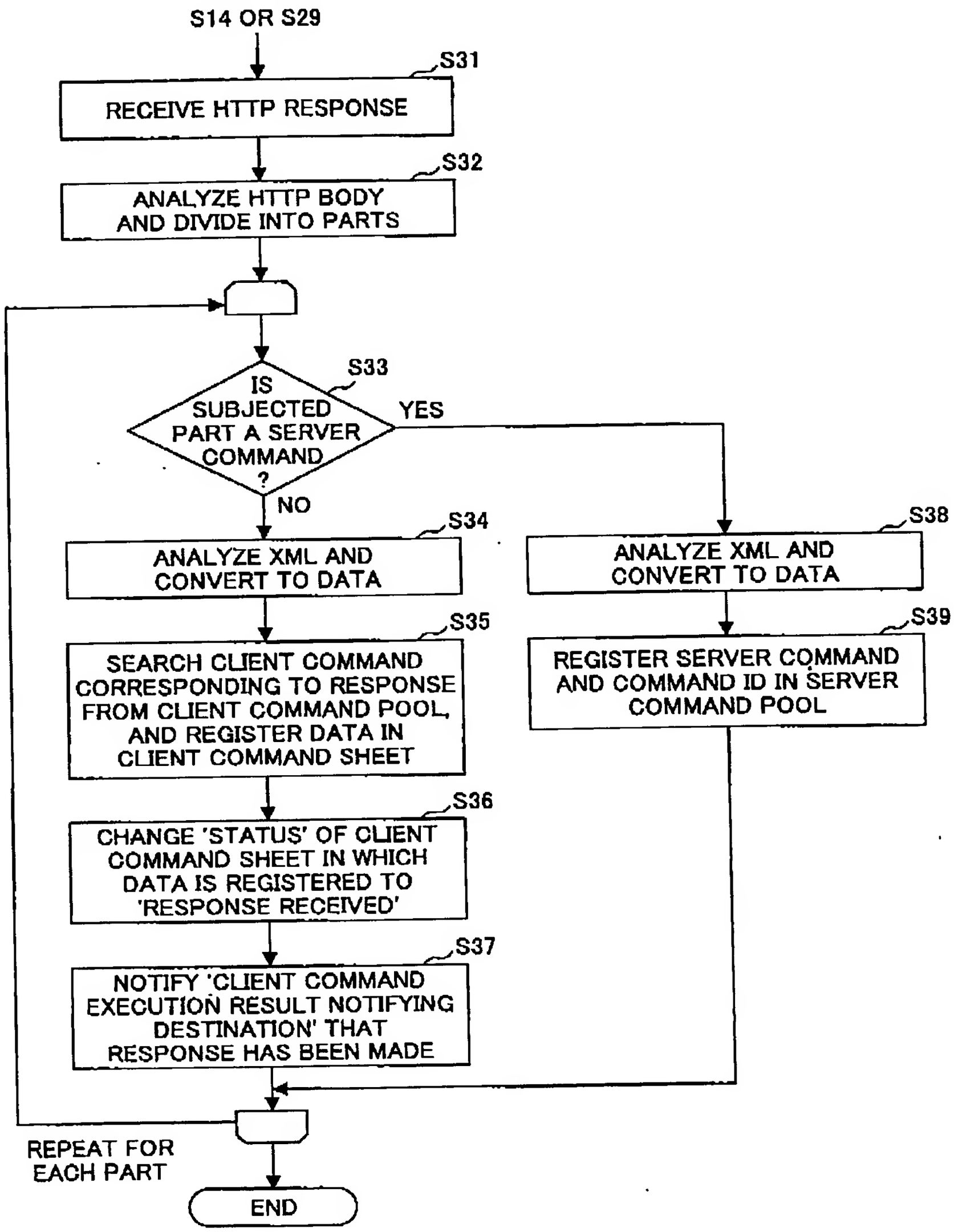

S14 OR S29
S31
RECEIVE HTTP RESPONSE
S32
ANALYZE HTTP BODY AND DIVIDE INTO PARTS
S33
IS SUBJECTED PART A SERVER COMMAND ?
YES
NO
S34
ANALYZE XML AND CONVERT TO DATA
S35
SEARCH CLIENT COMMAND CORRESPONDING TO RESPONSE FROM CLIENT COMMAND POOL, AND REGISTER DATA IN CLIENT COMMAND SHEET
S36
CHANGE 'STATUS' OF CLIENT COMMAND SHEET IN WHICH DATA IS REGISTERED TO 'RESPONSE RECEIVED'
S37
NOTIFY 'CLIENT COMMAND EXECUTION RESULT NOTIFYING DESTINATION' THAT RESPONSE HAS BEEN MADE
S38
ANALYZE XML AND CONVERT TO DATA
S39
REGISTER SERVER COMMAND AND COMMAND ID IN SERVER COMMAND POOL
REPEAT FOR EACH PART
END

FIG.20

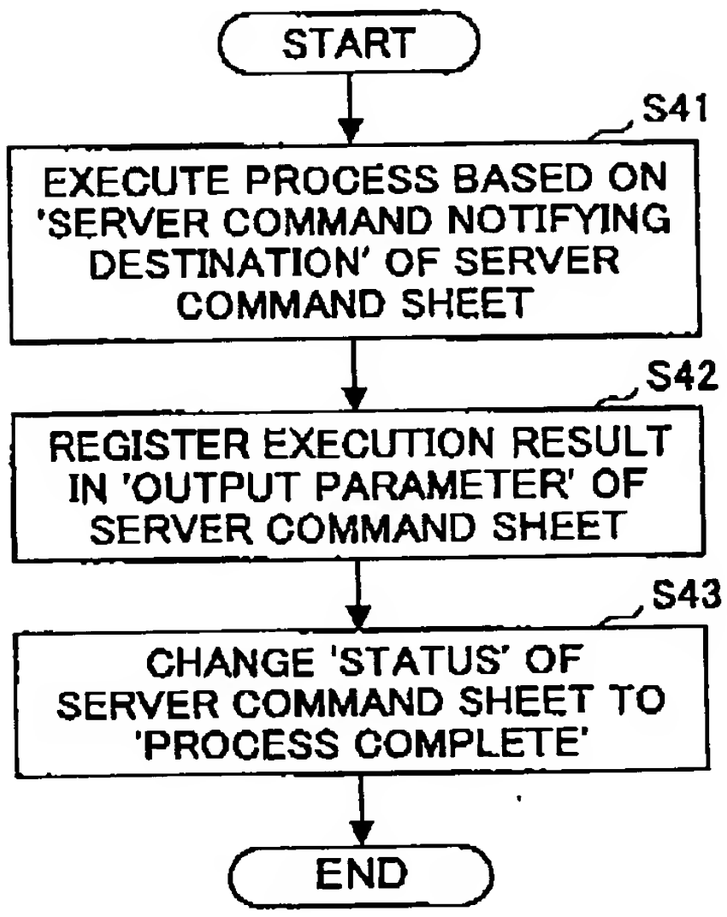

START
S41
EXECUTE PROCESS BASED ON 'SERVER COMMAND NOTIFYING DESTINATION' OF SERVER COMMAND SHEET
S42
REGISTER EXECUTION RESULT IN 'OUTPUT PARAMETER' OF SERVER COMMAND SHEET
S43
CHANGE 'STATUS' OF SERVER COMMAND SHEET TO 'PROCESS COMPLETE'
END

FIG.21

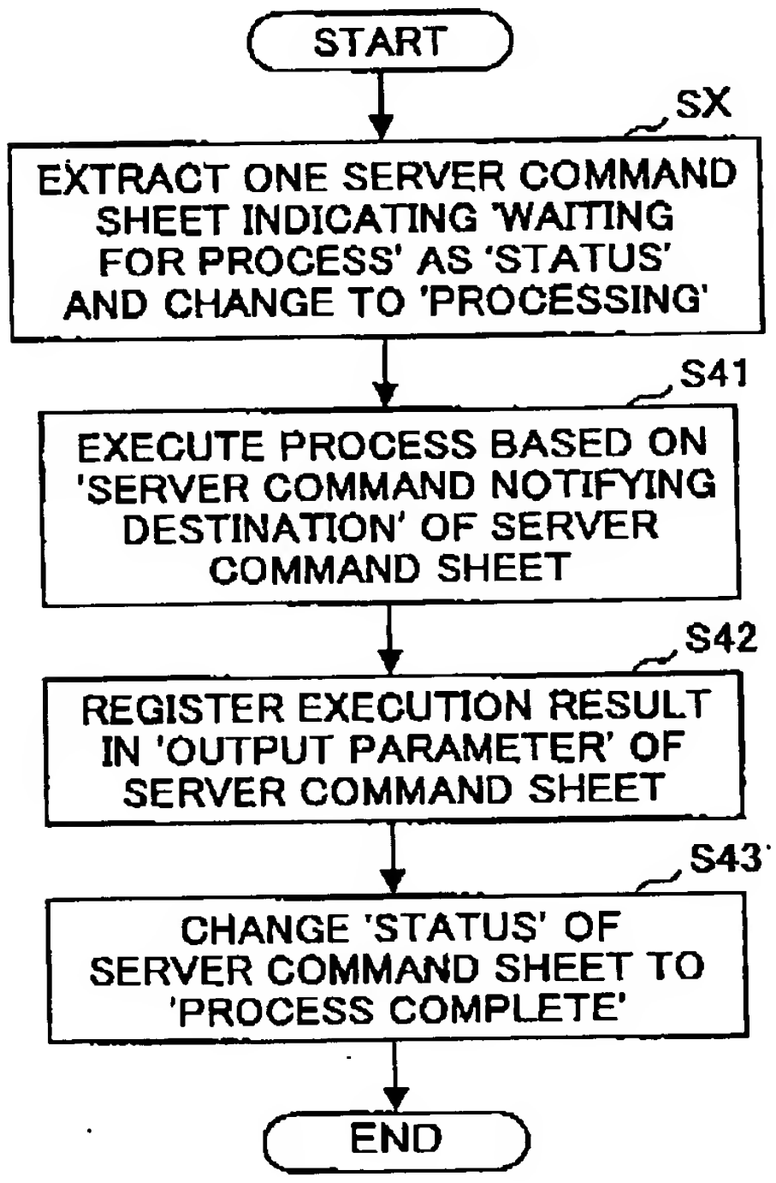
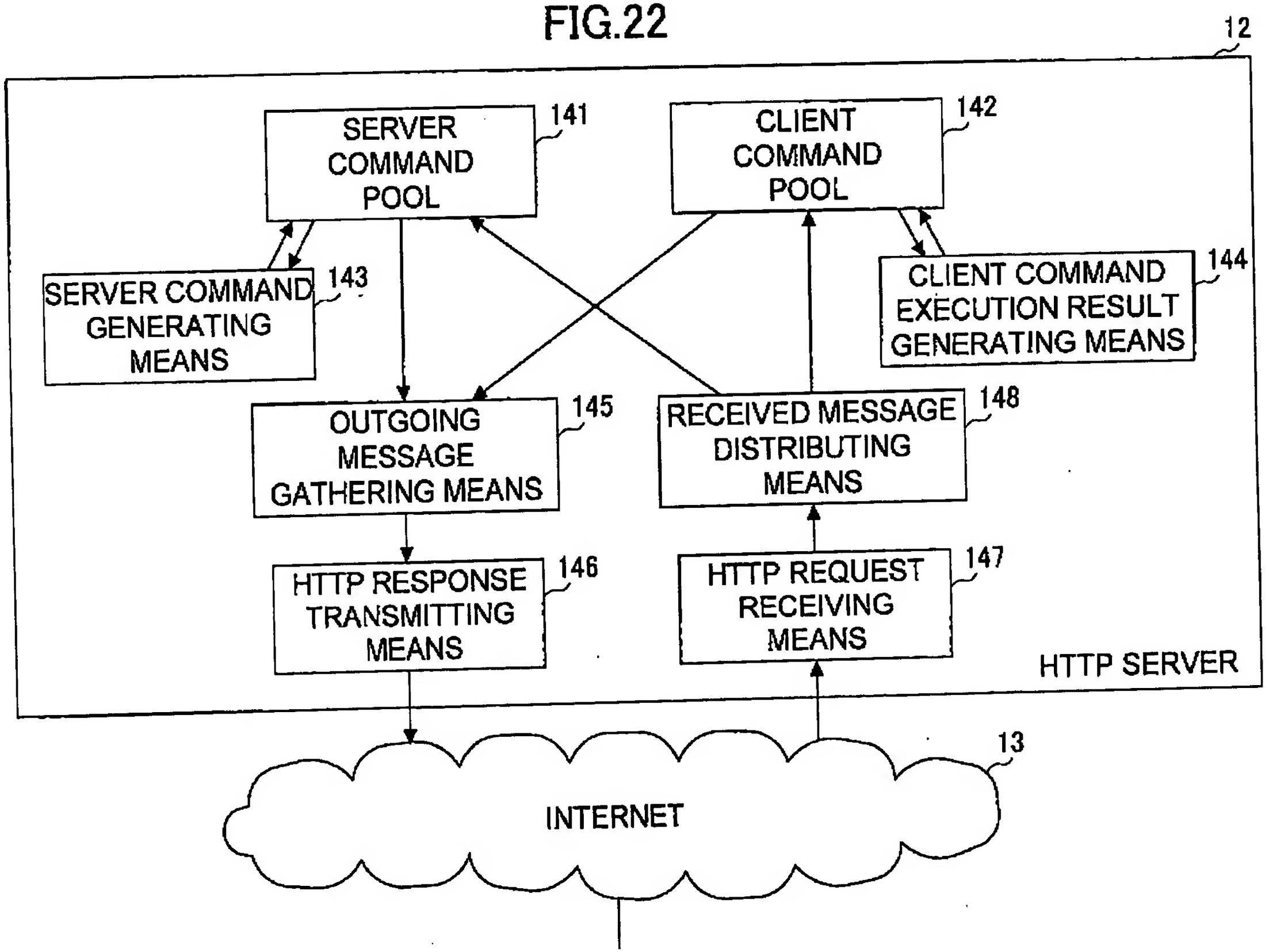

START
SX
EXTRACT ONE SERVER COMMAND SHEET INDICATING 'WAITING FOR PROCESS' AS 'STATUS' AND CHANGE TO 'PROCESSING'
S41
EXECUTE PROCESS BASED ON 'SERVER COMMAND NOTIFYING DESTINATION' OF SERVER COMMAND SHEET
S42
REGISTER EXECUTION RESULT IN 'OUTPUT PARAMETER' OF SERVER COMMAND SHEET
S43
CHANGE 'STATUS' OF SERVER COMMAND SHEET TO 'PROCESS COMPLETE'
END

FIG.22



OBLON, SPIVAK, ET AL
DOCKET #: 243019US2
INV: Hiroyuki MATSUSHIMA
SHEET 18 OF 31

FIG.23

SERVER COMMAND SHEET IN HTTP SERVER

COMMAND ID
METHOD NAME (e.g., TROUBLE NOTIFICATION)
INPUT PARAMETER (e.g., TROUBLE CONTENT)
STATUS (INITIAL VALUE: NOT PROCESSED)
SERVER COMMAND EXECUTION RESULT NOTIFYING DESTINATION
OUTPUT PARAMETER (BLANK UNTIL RESPONSE IS ACQUIRED)

FIG.24

CLIENT COMMAND SHEET IN HTTP SERVER

COMMAND ID
METHOD NAME (e.g., COUNTER ACQUISITION)
INPUT PARAMETER
STATUS (INITIAL VALUE: NOT PROCESSED)
OUTPUT PARAMETER (BLANK UNTIL PROCESS IS COMPLETED)
CLIENT COMMAND NOTIFYING DESTINATION

FIG.25

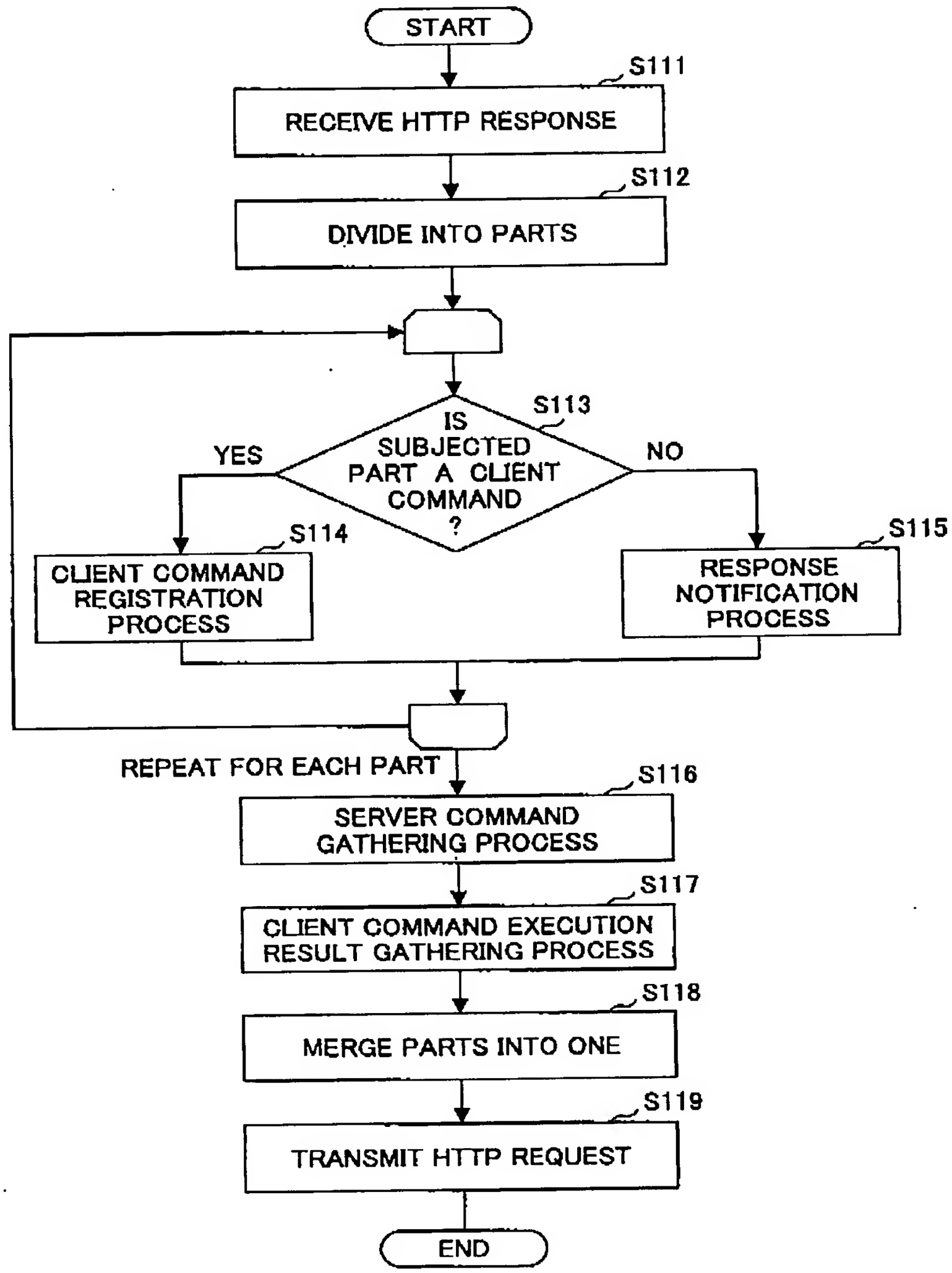

START
S111
RECEIVE HTTP RESPONSE
S112
DIVIDE INTO PARTS
S113
IS SUBJECTED PART A CLIENT COMMAND ?
YES
NO
S114
CLIENT COMMAND REGISTRATION PROCESS
S115
RESPONSE NOTIFICATION PROCESS
REPEAT FOR EACH PART
S116
SERVER COMMAND GATHERING PROCESS
S117
CLIENT COMMAND EXECUTION RESULT GATHERING PROCESS
S118
MERGE PARTS INTO ONE
S119
TRANSMIT HTTP REQUEST
END

FIG.26

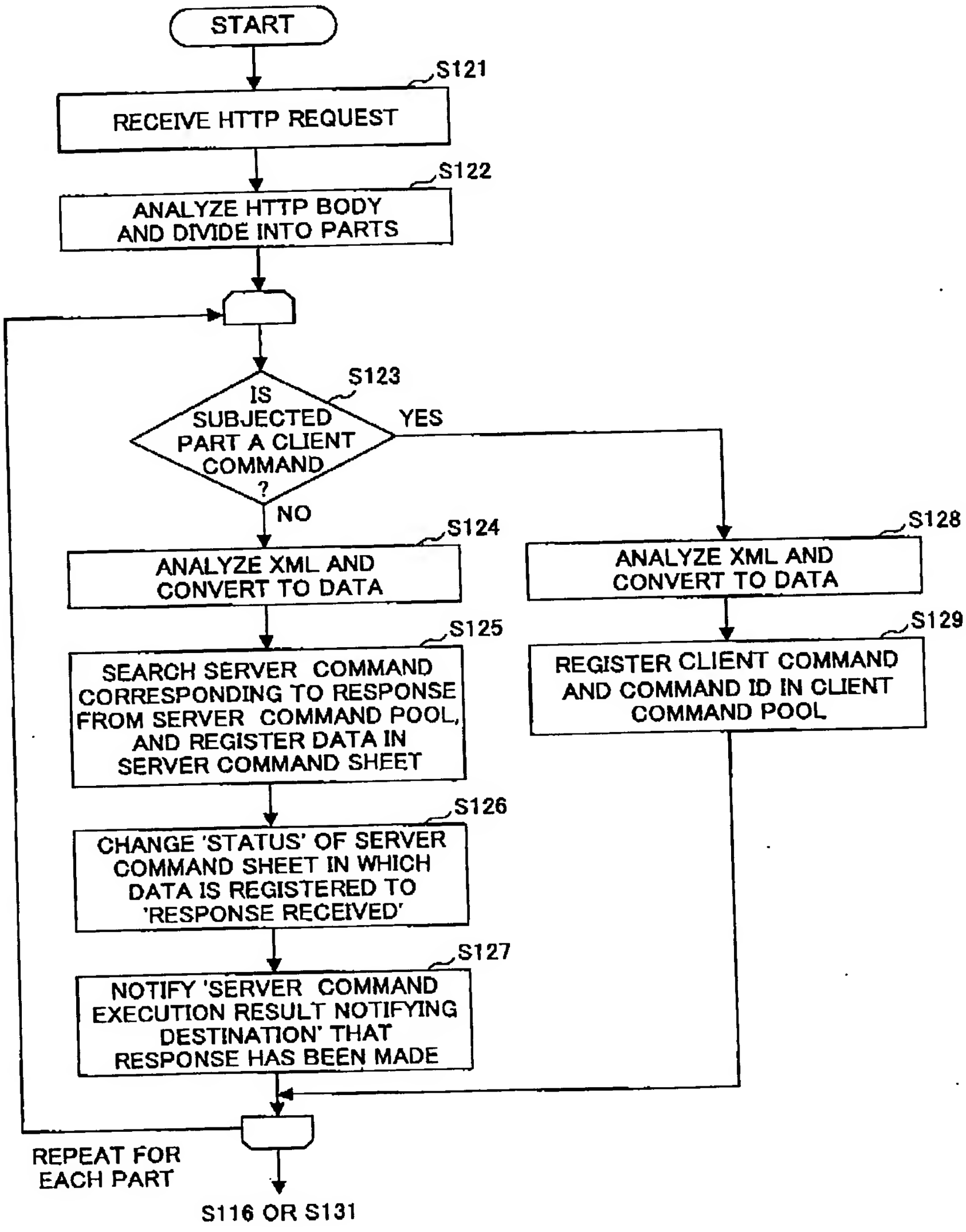

START
S121
RECEIVE HTTP REQUEST
S122
ANALYZE HTTP BODY AND DIVIDE INTO PARTS
S123
IS SUBJECTED PART A CLIENT COMMAND ?
YES
NO
S124
ANALYZE XML AND CONVERT TO DATA
S125
SEARCH SERVER COMMAND CORRESPONDING TO RESPONSE FROM SERVER COMMAND POOL, AND REGISTER DATA IN SERVER COMMAND SHEET
S126
CHANGE 'STATUS' OF SERVER COMMAND SHEET IN WHICH DATA IS REGISTERED TO 'RESPONSE RECEIVED'
S127
NOTIFY 'SERVER COMMAND EXECUTION RESULT NOTIFYING DESTINATION' THAT RESPONSE HAS BEEN MADE
S128
ANALYZE XML AND CONVERT TO DATA
S129
REGISTER CLIENT COMMAND AND COMMAND ID IN CLIENT COMMAND POOL
REPEAT FOR EACH PART
S116 OR S131

FIG.27

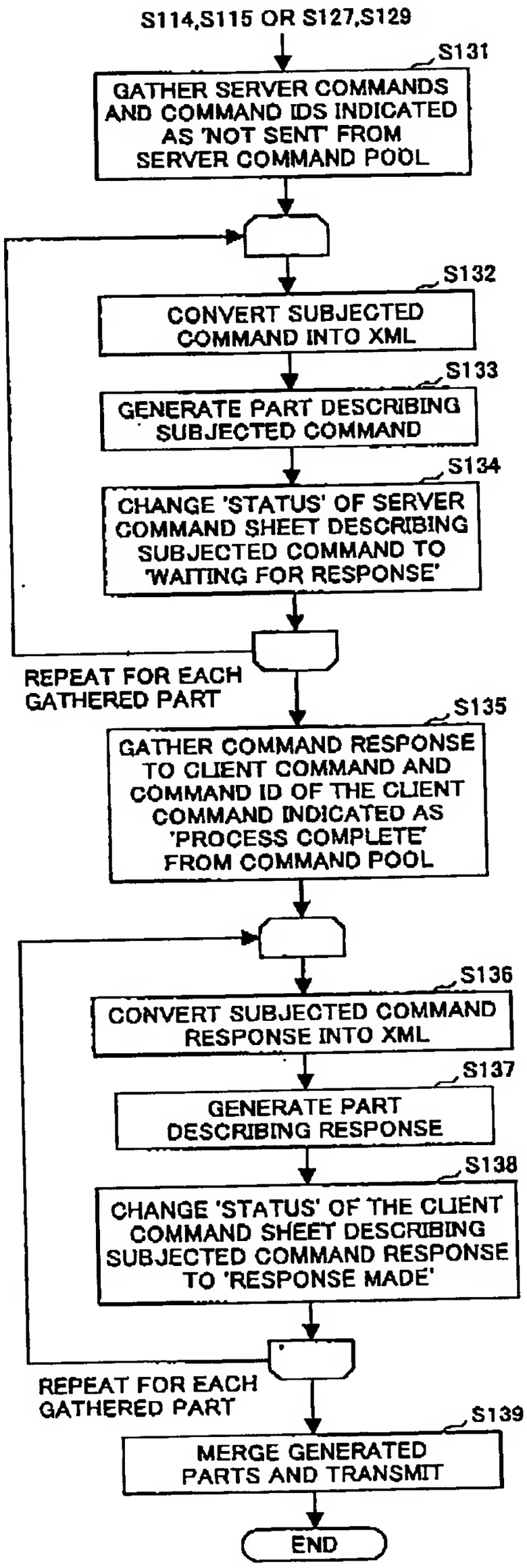

S114,S115 OR S127,S129
S131
GATHER SERVER COMMANDS AND COMMAND IDS INDICATED AS 'NOT SENT' FROM SERVER COMMAND POOL
S132
CONVERT SUBJECTED COMMAND INTO XML
S133
GENERATE PART DESCRIBING SUBJECTED COMMAND
S134
CHANGE 'STATUS' OF SERVER COMMAND SHEET DESCRIBING SUBJECTED COMMAND TO 'WAITING FOR RESPONSE'
REPEAT FOR EACH GATHERED PART
S135
GATHER COMMAND RESPONSE TO CLIENT COMMAND AND COMMAND ID OF THE CLIENT COMMAND INDICATED AS 'PROCESS COMPLETE' FROM COMMAND POOL
S136
CONVERT SUBJECTED COMMAND RESPONSE INTO XML
S137
GENERATE PART DESCRIBING RESPONSE
S138
CHANGE 'STATUS' OF THE CLIENT COMMAND SHEET DESCRIBING SUBJECTED COMMAND RESPONSE TO 'RESPONSE MADE'
REPEAT FOR EACH GATHERED PART
S139
MERGE GENERATED PARTS AND TRANSMIT
END

FIG.28

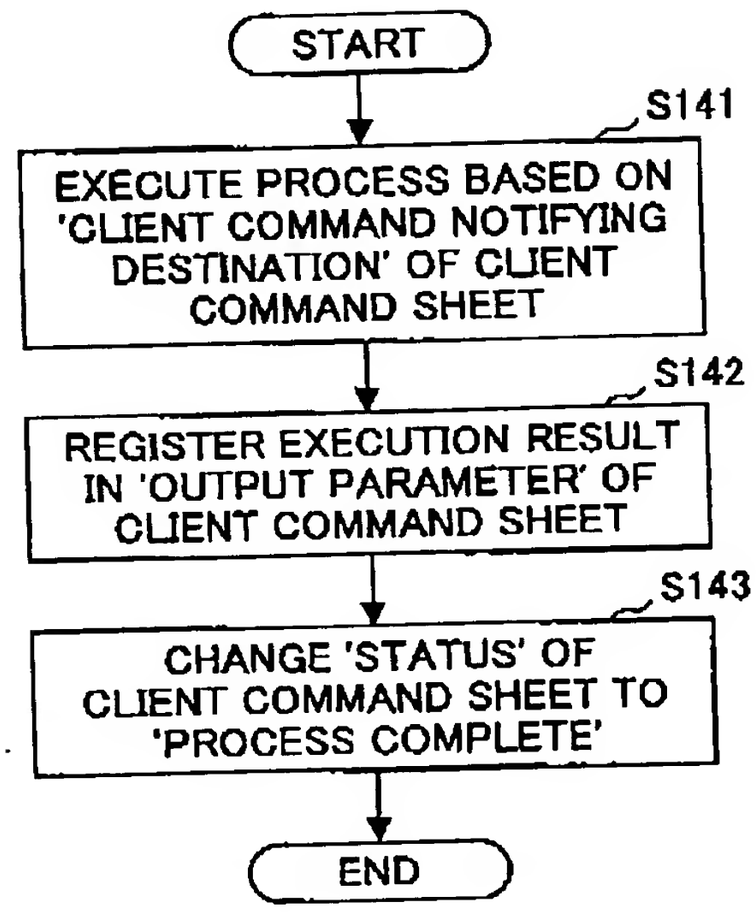

START
S141
EXECUTE PROCESS BASED ON 'CLIENT COMMAND NOTIFYING DESTINATION' OF CLIENT COMMAND SHEET
S142
REGISTER EXECUTION RESULT IN 'OUTPUT PARAMETER' OF CLIENT COMMAND SHEET
S143
CHANGE 'STATUS' OF CLIENT COMMAND SHEET TO 'PROCESS COMPLETE'
END

FIG.29

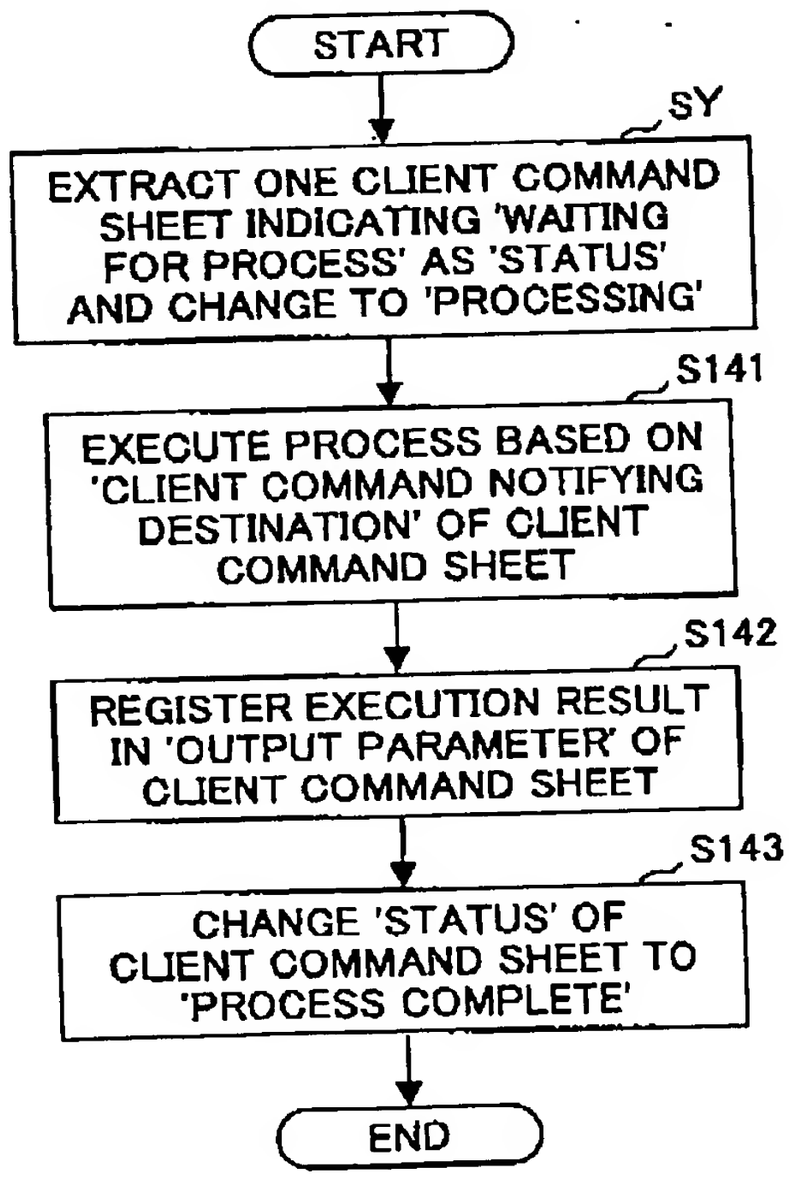

START
SY
EXTRACT ONE CLIENT COMMAND SHEET INDICATING 'WAITING FOR PROCESS' AS 'STATUS' AND CHANGE TO 'PROCESSING'
S141
EXECUTE PROCESS BASED ON 'CLIENT COMMAND NOTIFYING DESTINATION' OF CLIENT COMMAND SHEET
S142
REGISTER EXECUTION RESULT IN 'OUTPUT PARAMETER' OF CLIENT COMMAND SHEET
S143
CHANGE 'STATUS' OF CLIENT COMMAND SHEET TO 'PROCESS COMPLETE'
END

FIG.30

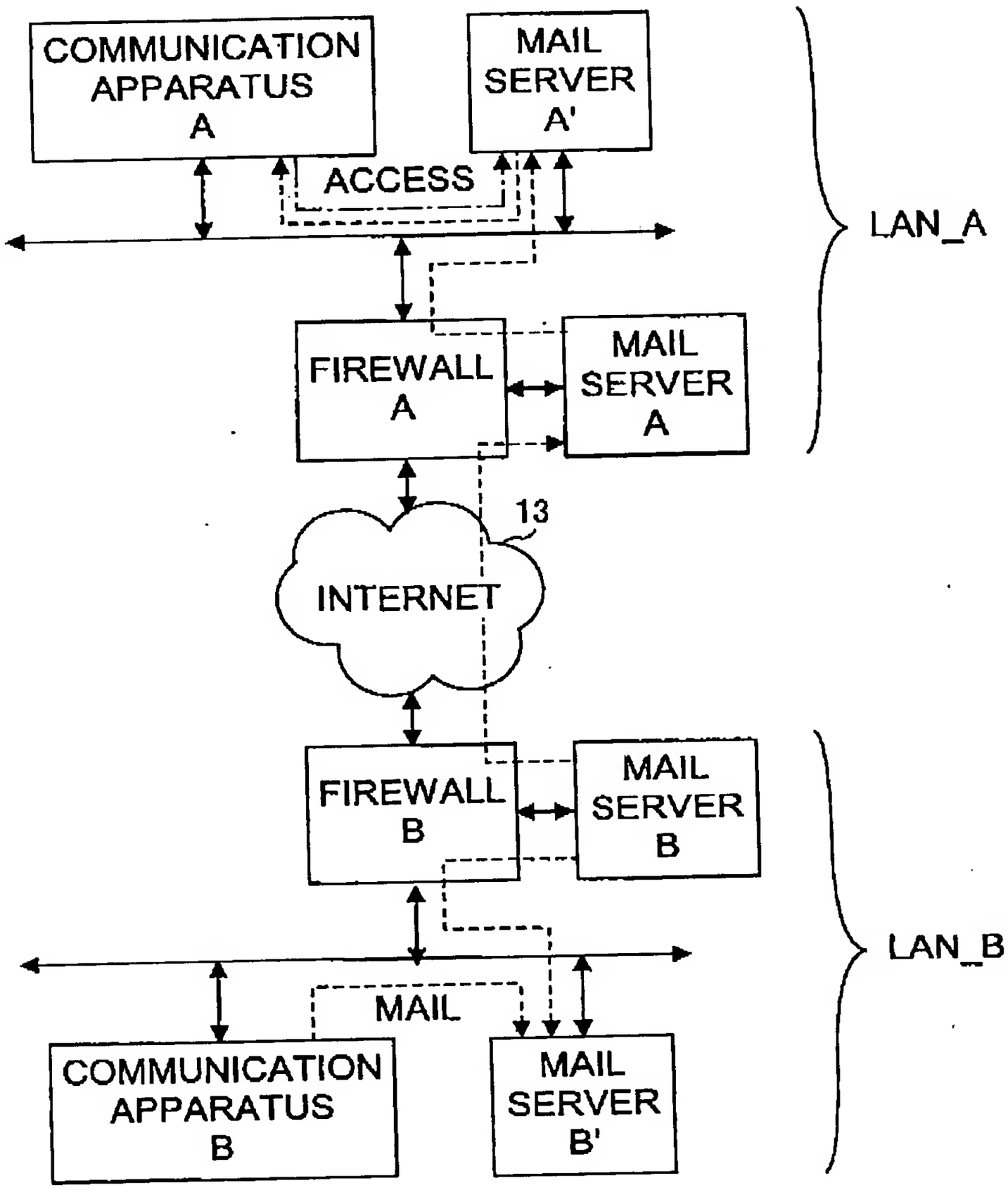

COMMUNICATION APPARATUS A
MAIL SERVER A'
ACCESS
LAN_A
FIREWALL A
MAIL SERVER A
13
INTERNET
FIREWALL B
MAIL SERVER B
LAN_B
MAIL
COMMUNICATION APPARATUS B
MAIL SERVER B'

FIG.31A

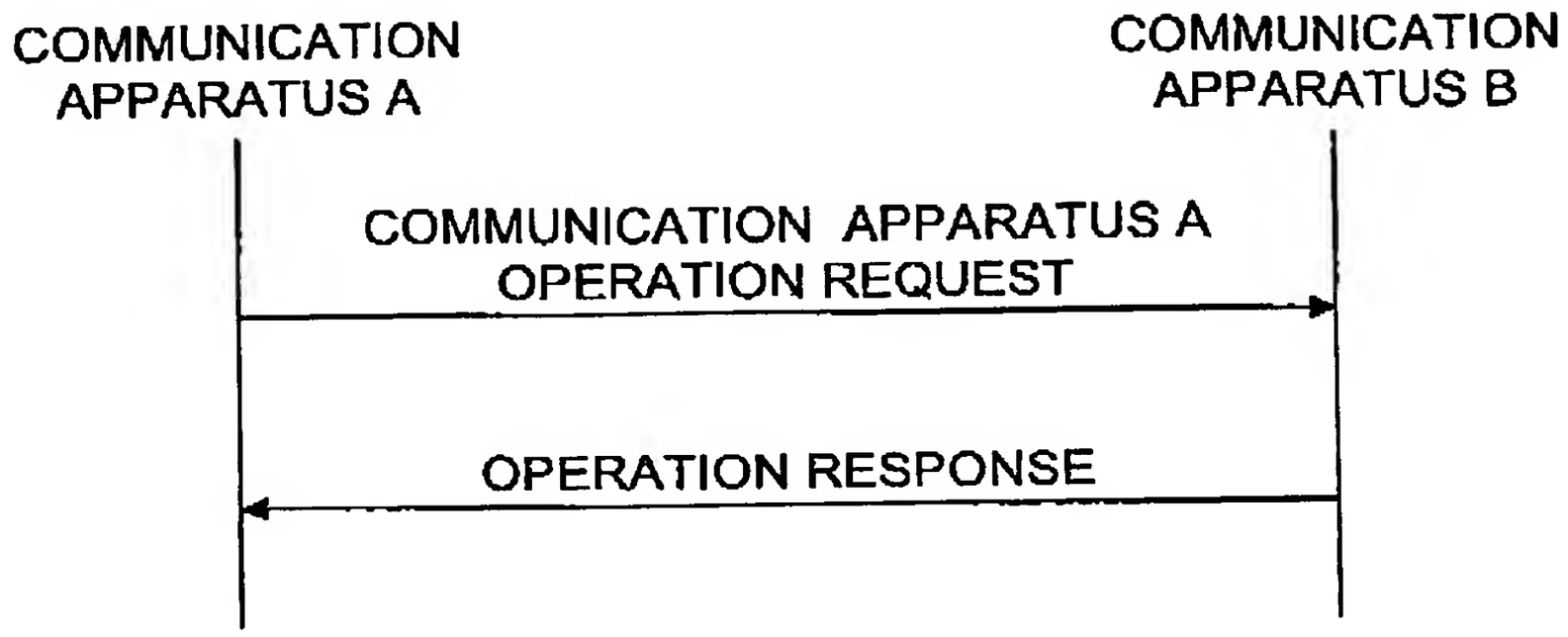


FIG.31B

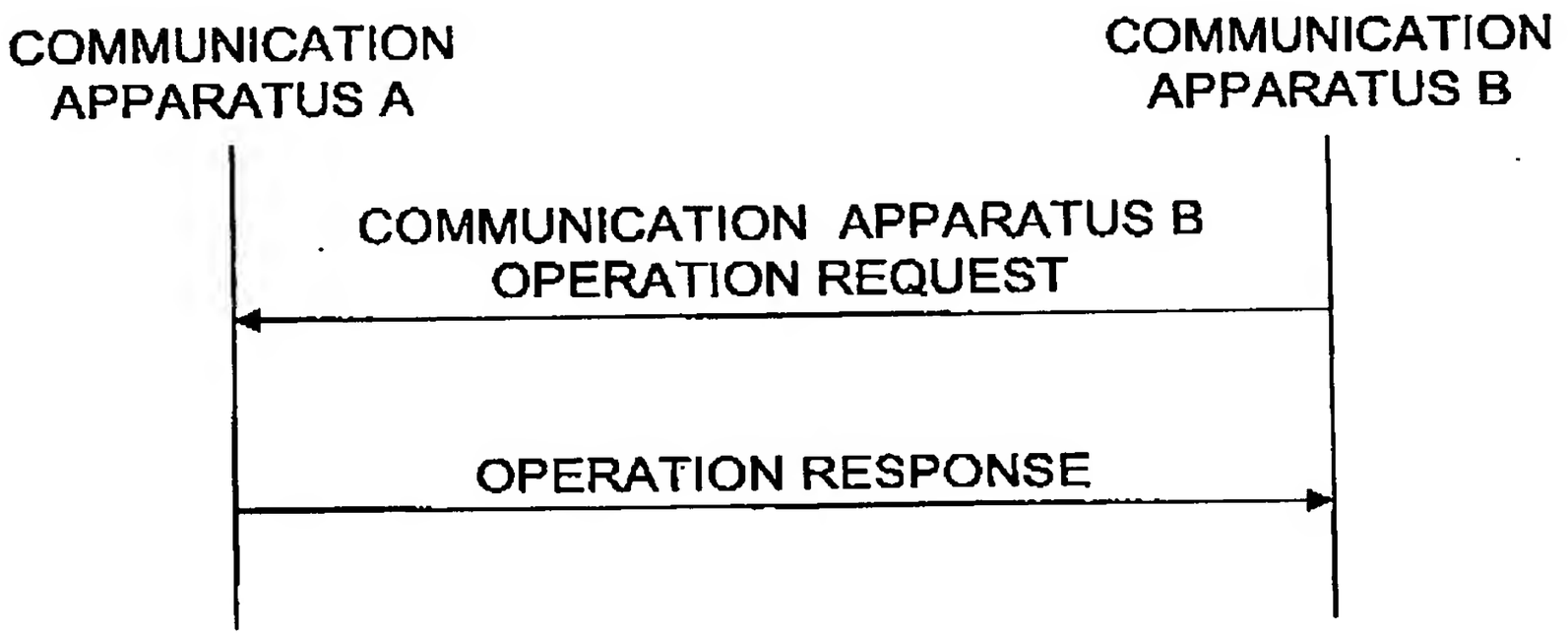


FIG.32

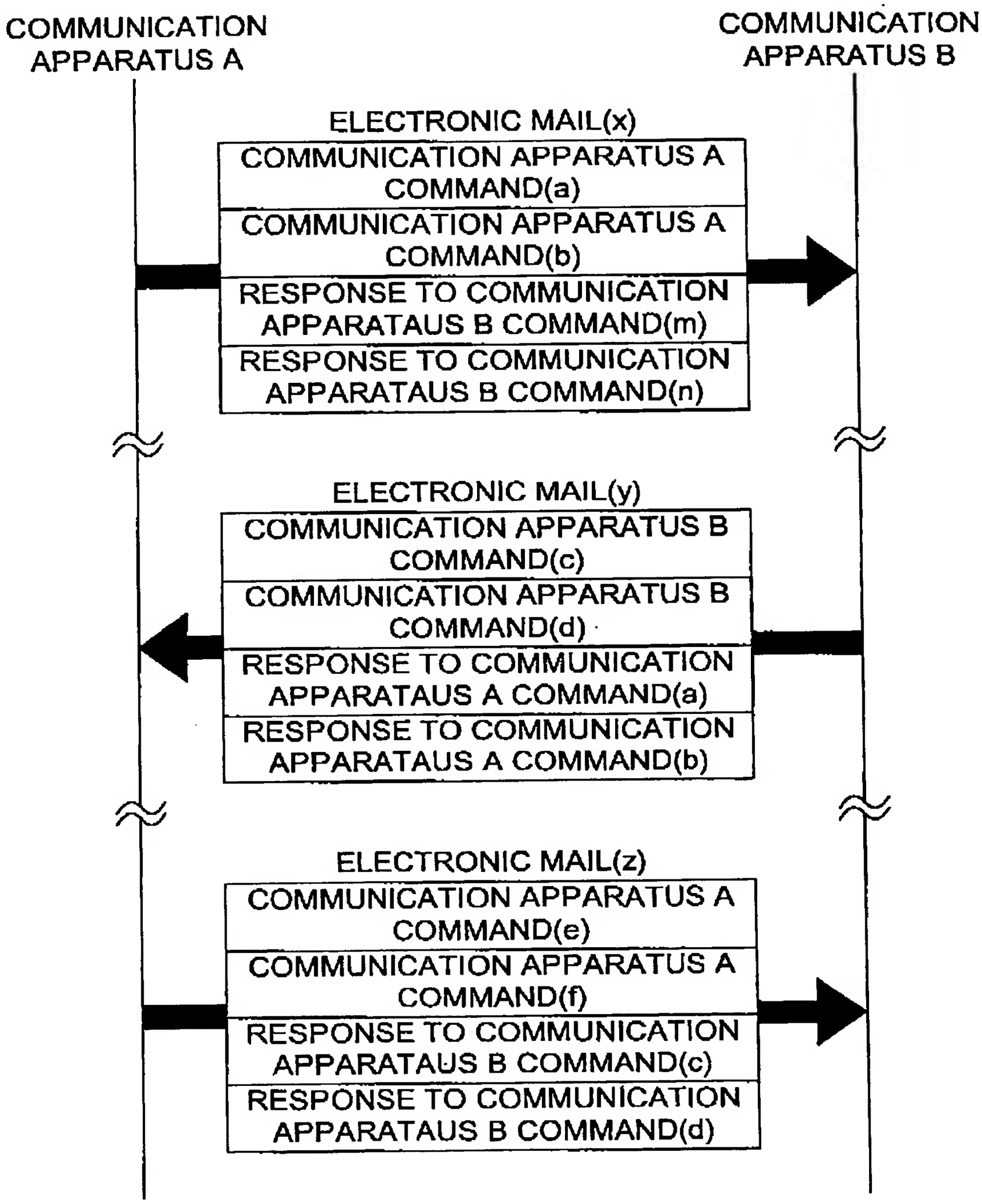
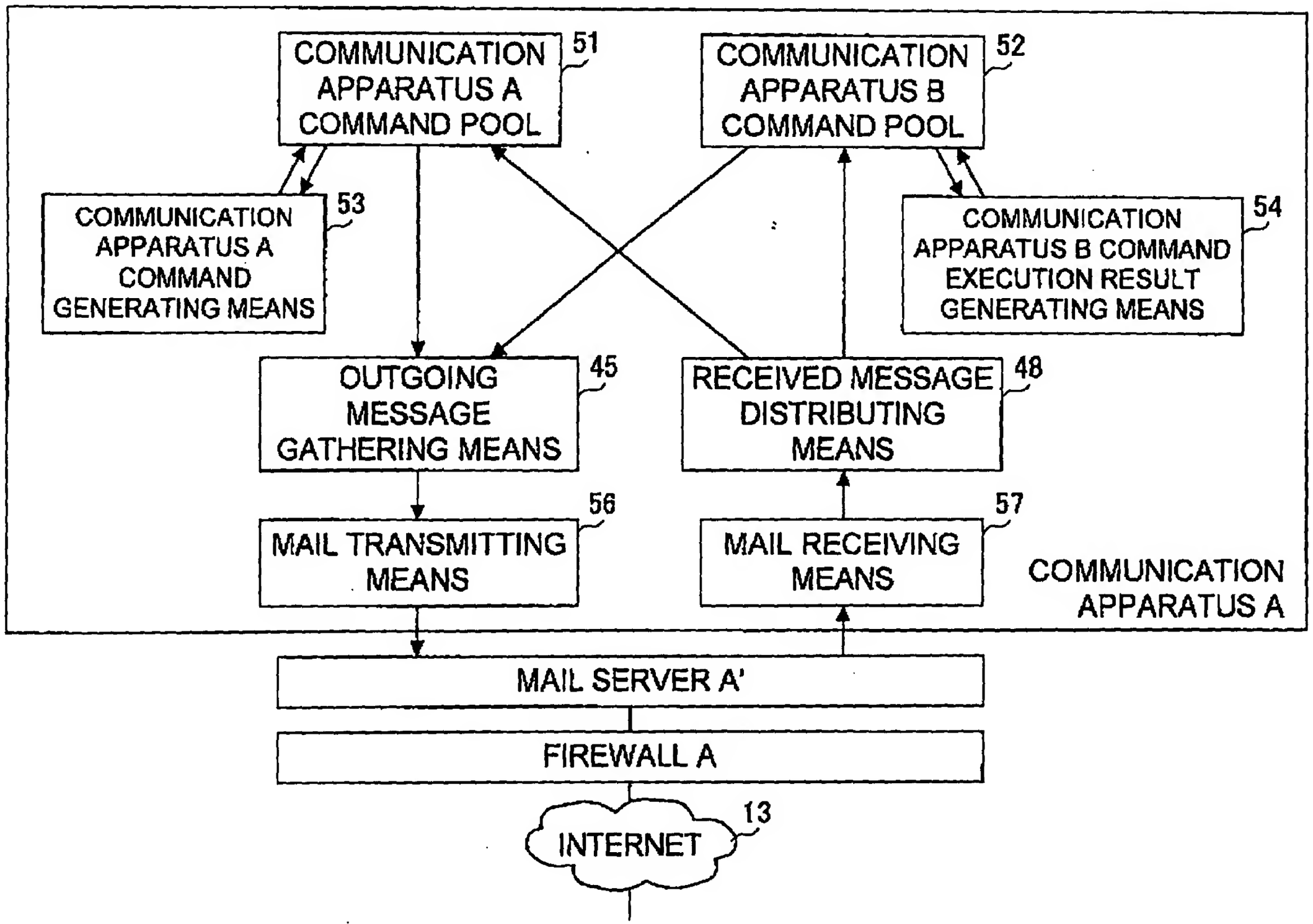

COMMUNICATION APPARATUS A
COMMUNICATION APPARATUS B
ELECTRONIC MAIL(x)
COMMUNICATION APPARATUS A COMMAND(a)
COMMUNICATION APPARATUS A COMMAND(b)
RESPONSE TO COMMUNICATION APPARATAUS B COMMAND(m)
RESPONSE TO COMMUNICATION APPARATAUS B COMMAND(n)
ELECTRONIC MAIL(y)
COMMUNICATION APPARATUS B COMMAND(c)
COMMUNICATION APPARATUS B COMMAND(d)
RESPONSE TO COMMUNICATION APPARATAUS A COMMAND(a)
RESPONSE TO COMMUNICATION APPARATAUS A COMMAND(b)
ELECTRONIC MAIL(z)
COMMUNICATION APPARATUS A COMMAND(e)
COMMUNICATION APPARATUS A COMMAND(f)
RESPONSE TO COMMUNICATION APPARATAUS B COMMAND(c)
RESPONSE TO COMMUNICATION APPARATAUS B COMMAND(d)

FIG.33



OBLON, SPIVAK, ET AL
DOCKET #: 243019US2
INV: Hiroyuki MATSUSHIMA
SHEET 27 OF 31

FIG.34

```
From: deviceA@foo.com
To: deviceB@bar.com
Subject: Multi Message #0000001
Date: Web, 30 Jul 2003 10:00:00  +0900
Content-Type:multipart/mixed;boundary=MIME_boundary
Content-Length:nnnn

--MIME_boundary
Content-Type:text/xml;charset=iso-2022-jp
Content-Transfer-Encoding:7bit
SOAPAction:"SOAP Action URI"
X-SOAP-Type:Requset

<s:Envelope>
   <--SOAP Request-->
</s:Envelope>

--MIME_boundary
Content-Type:text/xml;charset=iso-2022-jp
Content-Transfer-Encoding:7bit
X-SOAP-Type:Response

<s:Envelope>
   <--SOAP Response-->
</s:Envelope>

--MIME_boundary
Content-Type:text/xml;charset=iso-2022-jp
Content-Transfer-Encoding:7bit
X-SOAP-Type:Response

<s:Envelope>
   <--SOAP Response-->
</s:Envelope>

--MIME_boundary
Content-Type:text/xml;charset=iso-2022-jp
Content-Transfer-Encoding:7bit
X-SOAP-Type:Response

<s:Envelope>
   <--SOAP Response-->
</s:Envelope>

--MINE_boundary--
```

PART 1

PART 2

PART 3

PART 4

FIG.35

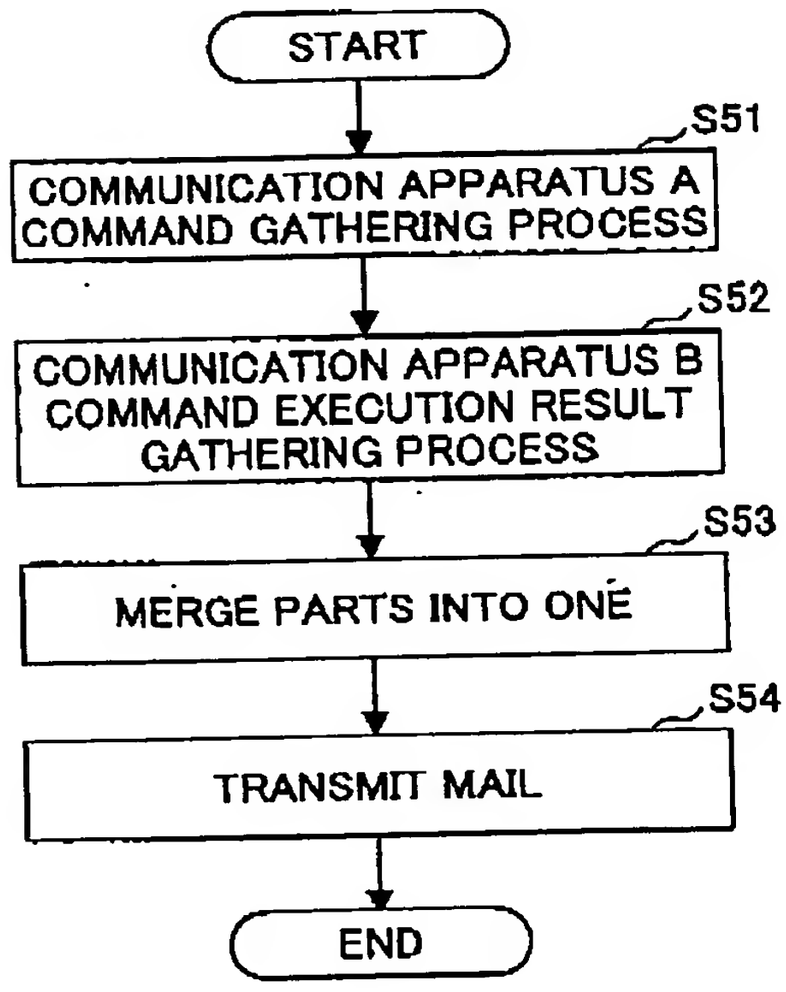

START
S51
COMMUNICATION APPARATUS A COMMAND GATHERING PROCESS
S52
COMMUNICATION APPARATUS B COMMAND EXECUTION RESULT GATHERING PROCESS
S53
MERGE PARTS INTO ONE
S54
TRANSMIT MAIL
END

FIG.36

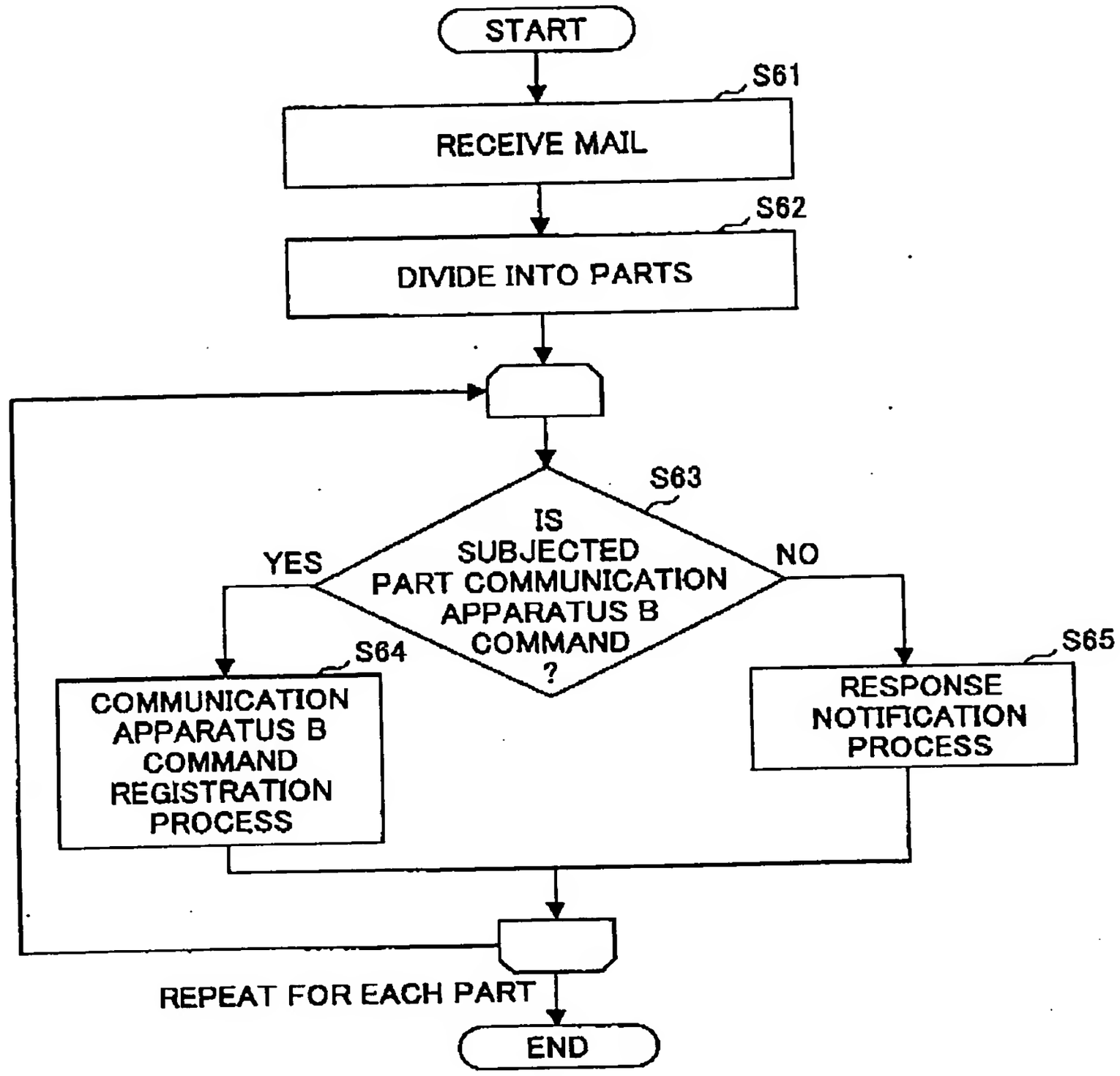
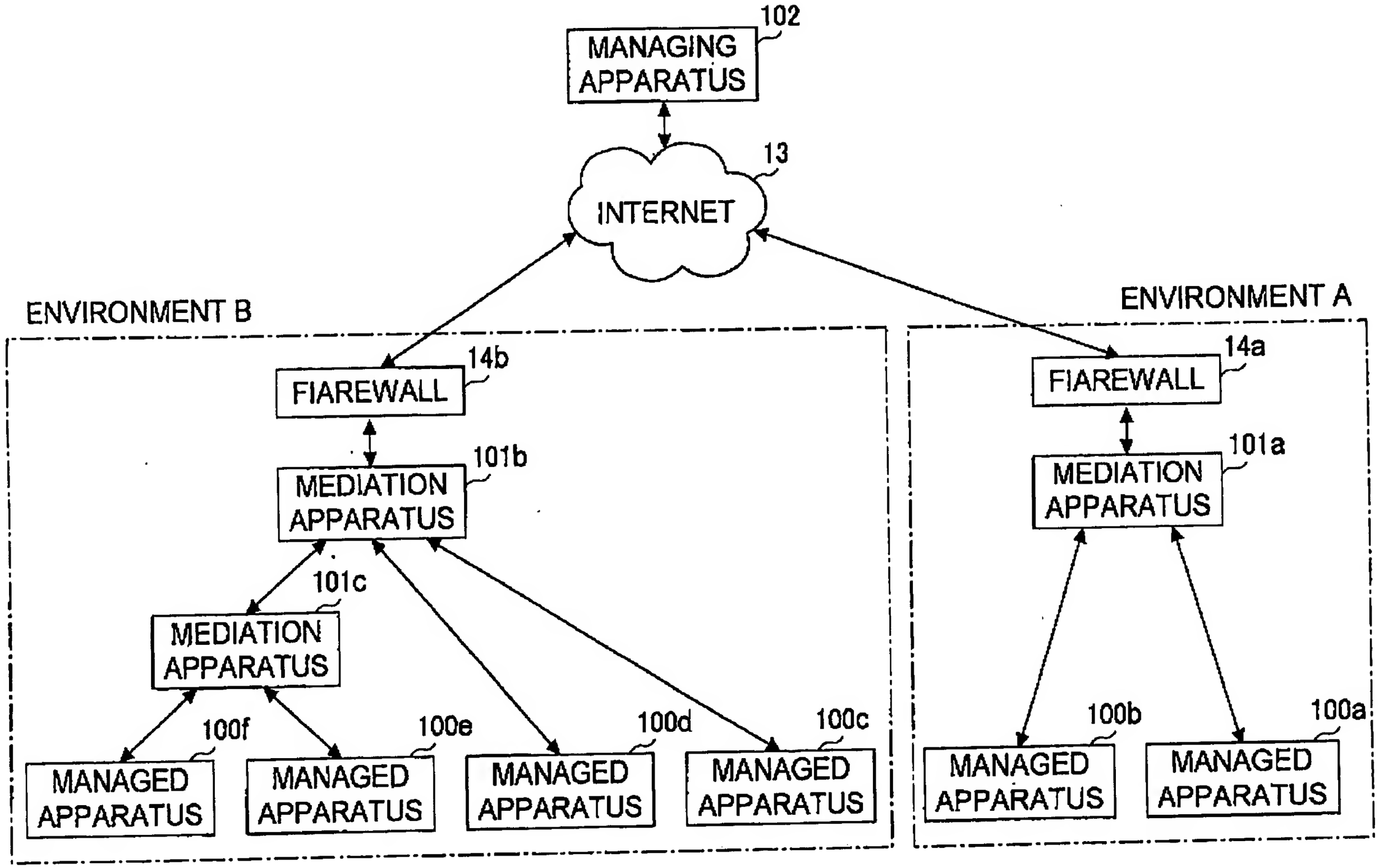

START
S61
RECEIVE MAIL
S62
DIVIDE INTO PARTS
S63
IS SUBJECTED PART COMMUNICATION APPARATUS B COMMAND ?
YES
NO
S64
COMMUNICATION APPARATUS B COMMAND REGISTRATION PROCESS
S65
RESPONSE NOTIFICATION PROCESS
REPEAT FOR EACH PART
END

FIG.37



OBLON, SPIVAK, ET AL
DOCKET #: 243019US2
INV: Hiroyuki MATSUSHIMA
SHEET 31 OF 31